



**Universitat Autònoma
de Barcelona**

Sistema Web para compartir plazas de parking

Memoria del proyecto
De Ingeniería Técnica en
Informática de Gestión
Realizado por
Daniel Martínez Domingo
y dirigido por
Marta Prim Sabrià

Escuela Universitaria de Informática
Sabadell, junio de 2009

Resumen del proyecto

El proyecto “Sistema Web para compartir plazas de parking” nace para conseguir una movilidad sostenible y segura en nuestros municipios.

Aparcar en las ciudades ha pasado a ser un problema grave que genera, además del estrés de no encontrar dónde dejar el vehículo, graves problemas de congestión de la vía pública.

El proyecto da la posibilidad a las personas que tienen una plaza de aparcamiento y que durante el día, la noche, fines de semana o en un horario variable la dejan libre para ir al trabajo o para estudiar fuera de los municipios, la puedan intercambiar, alquilar o vender con otras personas que venga a trabajar y/o estudiar a la ciudad donde se encuentra la plaza.

En este proyecto se presenta un buscador inteligente que lista las plazas por proximidad a la calle buscada y la georeferencia en un mapa para saber exactamente la localización, de esta manera es muy sencillo decidir si la plaza de aparcamiento está próxima al lugar al cual nos desplazamos.

Además el servicio Web da la posibilidad de subir fotografías de las plazas para más detalle.

También se dota al usuario de contratos legales para que no tenga que preocuparse de aspectos legales y tenga la certeza que la ley lo ampara.

Dado que el sistema Web está pensado para la administración pública cumple las normativas de accesibilidad a nivel AA, así como los test de validación de xHtml y Css.

A más a más se puede integrar perfectamente en las páginas de los clientes sin que se note que se está utilizando un servicio externo.

Índice

Capítulo 1: Introducción	
1.1 Introducción	1
1.2 Descripción del proyecto	1
1.3 Descripción de objetivos	2
1.4 Estado del arte	2
1.5 Motivaciones	3
1.6 Estructura de la memoria	3
Capítulo 2: Planificación y desarrollo del proyecto	
2.1 Introducción	5
2.2 Situación actual	5
2.3 Perfil del cliente – usuario	5
2.4 Objetivos	6
2.5 Fuentes de información	7
2.6 Descripción	7
2.7 Definición de etapas y metodología de desarrollo	8
2.8 Modelo de desarrollo	9
2.9 Recursos	9
2.10 Análisis coste – beneficio	11
2.11 Presupuesto	12
2.12 Evaluación de riesgos	13
2.13 Alternativas	14
2.14 Planificación	16
2.15 Viabilidad legal	18
2.16 Conclusiones	19
Capítulo 3: Requerimientos funcionales y no funcionales	
3.1 Estudio de la Base de datos	20
3.2 Requerimientos funcionales	29
3.3 Requerimientos no funcionales	33
Capítulo 4: Interficie	
4.1 Interficie de la aplicación	38
4.2 Integración en la Web del cliente	52
Capítulo 5: Pruebas	
5.1 Buscar plaza	55
5.2 Resultados de la búsqueda	57
5.3 Crear y/o modificar plaza	60
5.4 Registrarme y mis datos	64
5.5 Baja de usuario	67
5.6 Entrar	68
5.7 Contraseña olvidada	69
5.8 Recuperar contraseña olvidada	70
5.9 Mis plazas	72
5.10 Mis mensajes	75
5.11 Administración, nuevo cliente y/o modificar cliente	76
5.12 Díselo a un amigo	77
5.13 Errores de funcionamiento	78
5.14 Comprobaciones generales	79

Capítulo 6: Codificación	
6.1 Distribución de los ficheros	81
6.2 Ficheros .htaccess	81
6.3 Ficheros de configuración	82
6.4 FrontController	82
6.5 Objetos	82
6.6 Alias	82
6.7 Controladores	82
6.8 Modelos	83
6.9 Vista	83
6.10 Utilidades	84
6.11 Seguridad	85
6.12 Personalización para un cliente	85
6.13 Crear un módulo nuevo	86
6.14 Patrones utilizados	87
6.15 Problemas surgidos	88
Capítulo 7: Migración	
7.1 Migración de la Base de datos	98
7.2 Migración de los ficheros de texto	98
Capítulo 8: Conclusión	
8.1 Conclusión	99
8.2 Líneas futuras	100
Bibliografía	102
Anexo 1: Manual de usuario	
Anexo 2: Manual del programador	
Anexo 3: Diagrama de la Base de datos	
Anexo 4: Diagrama de casos de uso	
Anexo 5: Diagrama del flujo de la aplicación	
Anexo 6: Diagrama UML	

Capítulo

1

Introducción

1.1 Introducción

Después de pensar varios títulos como Compartir Parking, Compartir plazas de aparcamiento, etc. se ha decidido poner Sistema Web para compartir plazas de parking, ya que, dicho enunciado afirma de manera clara y concisa las aspiraciones y objetivos del proyecto.

Apostar por un proyecto competitivo y moderno es una de nuestras prioridades, que mejor manera de conseguirlo que orientándolo a las necesidades de una empresa tangible del siglo XXI.

La colaboración con una empresa que invierte en nuevas tecnologías y sabe las tendencias del mercado nos facilita la consecución de nuestros objetivos, así como la creación de un sistema moderno, dinámico y adaptado a nuestros días.

La empresa colaboradora en este proyecto es Compartir SL.

1.2 Descripción del proyecto

Se creará un servicio Web adaptado a las necesidades actuales de la sociedad, dando una posible solución al problema de aparcamiento.

Si durante el período de vacaciones, el día, la noche o una franja horaria en concreto una persona no utiliza su plaza de parking tendrá la opción de alquilarla, intercambiarla e incluso venderla.

Esto último, en el caso que dicha persona sea la propietaria de la plaza de aparcamiento y por cualquier motivo ya no la requiera.

El precio y las condiciones las pone el propietario.

En el caso que el poseedor de la plaza de aparcamiento no esté durante un periodo de tiempo en su plaza ya sea rutinario (horario fijo durante la semana) o esporádico (meses de vacaciones o fines de semana en concreto) existe la posibilidad de alquilar o intercambiar las plazas.

En el caso de alquiler el poseedor fijará los precios con el alquilante.

En el caso de que ambos poseedores de plazas coincidan en horarios y decidan intercambiar sus plazas.

En el mencionado proceso la empresa desarrolladora del sistema no obtiene ningún tipo de tasa ni beneficio de dichas transacciones.

El servicio Web de la empresa proporciona un sistema interno de mensajería para la comunicación entre vendedor/comprador, alquilante/alquilador, etc.

Dicho sistema posibilita el intercambio de teléfonos para una comunicación más directa y fluida.

1.3 Descripción de objetivos

El objetivo es crear un servicio Web capaz de agilizar y facilitar el intercambio, venta o alquiler de plazas de aparcamiento entre distintas personas.

El servicio es capaz de listar por proximidad las plazas más cercanas al lugar de destino, gestionar usuarios y plazas, entre otras funcionalidades.

1.4 Estado del arte

En el año 2000 a raíz de los problemas de aparcamiento que sufrían las ciudades se creó la empresa Compartir SL, la cual creó un servicio Web (<http://www.compartirparking.com/>) para compartir plazas de aparcamiento.

Después de 4 años apareció Plazas-garaje (<http://www.plazas-garaje.com/>) y unos años después BuscoParkings (<http://www.buscoparkings.com/>) ambos servicios Web ofrecen lo mismo que CompartirParking.

En este último año han aparecido otras dos empresas, Argaraz Servicios Inmobiliarios S.L. (<http://www.aparcanet.es/>) y aParcalia (<http://www.aparcalia.com/>) que ofrecen servicios idénticos a las anteriormente mencionadas. Todas estas empresas ofrecen su servicio Web de manera totalmente gratuita.

Actualmente la gestión de estas plazas está muy dividida y diversificada de manera que es difícil encontrar un sistema eficiente.

Podemos destacar algunas de las características de las empresas de las anteriores:

1. Compartir SL: esta organización trabaja exclusivamente para Ayuntamientos y grandes empresas. Adapta el servicio Web a la Web municipal o de la empresa para que estas entidades u organismos puedan ofrecérselo a sus ciudadanos/trabajadores. El servicio Web que ofrece se llama CompartirParking.

La potencia de CompartirParking es su Base de Datos, que une ciudadanos de varias localizaciones de la península incrementando el encuentro de coincidencias de plazas de aparcamiento entre ciudades.

Actualmente el servicio Web de CompartirParking está obsoleto y necesita una remodelación total, de aquí surge la iniciativa de este proyecto final de carrera.

2. aParcalia: es un servicio nuevo en este mercado pero cuenta con el apoyo del Ministerio de Industria (CDTI), Madridinnova, la Consejería de economía e innovación tecnológica de la Comunidad de Madrid y la Universidad Pontificia de Comillas.

3. Plazas-garaje y BuscoParkings: tienen un número considerable de plazas publicadas.

1.5 Motivaciones

Actualmente uno de los problemas que atañe a la inmensa mayoría de personas es el aparcamiento.

Tanto organismos públicos como empresas privadas intentan ingeniar una solución viable y económica para paliar el incremento exponencial del número de vehículos que circulan día a día por las ciudades.

En este proyecto planteamos una solución localizada y adaptada a las necesidades de cada municipio para la reducción de parte de dicho problema.

Nuestras motivaciones son la mejora de la sociedad actual mediante un sistema que facilite y descongestione las calles de vehículos.

1.6 Estructura de la memoria

La memoria se divide en 10 capítulos:

- El primer capítulo es el de la introducción, se explica el título y tipología, se define el proyecto y los objetivos que se cumplirán así como un breve análisis de otras aplicaciones ya existentes que cumplen parte de los objetivos de este proyecto. También hay un subapartado en el cual se explica el motivo que nos lleva a desarrollar esta aplicación Web.
- El segundo capítulo es el de planificación y desarrollo del proyecto, en éste se explica como planificar las tareas, tiempo y recursos. Además consta de un informe de viabilidad técnica así como uno de viabilidad económica. Los métodos de desarrollo también son expuestos en este apartado.
- El tercer capítulo es el de requisitos funcionales y no funcionales, en el cual se define la estructura modular de la aplicación y la estructura de datos. Por otro lado se explica el diseño de la Base de Datos así como una descripción de las clases y funciones.
- El cuarto capítulo es el de interfícies, en dicho capítulo se explican las diferentes pantallas que la aplicación mostrará. Se razonará la distribución, el orden, el color y los tamaños entre otros elementos visuales.
- El quinto capítulo es el de pruebas, se realizan todo tipo de pruebas tanto generales de nuestro servicio Web como de integración en otros servicios Web ya existentes. También se revisa el resultado final en diferentes navegadores

Web como IE6, IE7 y Mozilla Firefox. Para acabar se aplica el test de accesibilidad Web cuyo resultado tiene que ser positivo.

- El sexto capítulo es el de codificación, aquí se explican las diferentes partes del proyecto así como las cosas fundamentales para su comprensión y ampliación, también se detallan algunos de los problemas que han surgido durante el desarrollo además se ofrece una solución a dichos problemas en caso de haberlas encontrados o posibles soluciones en un futuro no muy lejano.
- El séptimo capítulo es el de migración, en este capítulo se detallan las acciones a realizar para que la migración tanto de ficheros como de la base de datos del sistema anterior pueda funcionar correctamente en la nueva aplicación.
- El capítulo octavo es el de conclusiones, aquí se expone de manera clara y sistemática un pequeño resumen de los problemas, restricciones y otros factores que han condicionado el proyecto. También se exponen un conjunto de ideas para ampliar o mejorar las funcionalidades de la aplicación. Para finalizar se describen una serie de valoraciones tanto objetivas como subjetivas del trabajo desarrollado.
- Por último tenemos la Bibliografía donde se listan todas las fuentes de información utilizadas.

Capítulo 2

Planificación y desarrollo del proyecto

2.1 Introducción

Con el proyecto se pretende paliar y dar una solución a uno de los problemas de la sociedad como es el aparcamiento municipal.

Las ventajas que nos brinda el proyecto son:

1. Información centralizada.
2. Fácil acceso a la información.
3. Rapidez en la búsqueda de plazas de aparcamiento.
4. Visualización de la plaza en un mapa.
5. Comunicación dinámica por mensajería interna.
6. Estadísticas y contratos automatizados.

Y los inconvenientes:

1. Formación de los trabajadores para la administración del nuevo servicio Web.
2. Familiarización con el nuevo sistema por parte de los usuarios.
3. Necesidad de mantenimiento más técnico.

2.2 Situación actual

Actualmente el cliente cuenta con un servicio Web similar al que se tiene que desarrollar.

El servicio que tiene está obsoleto y su adaptación tanto de código como de estilos es inviable dado que no sigue ninguna estructura, los contenidos no son accesibles y utiliza funciones actualmente en desuso en las nuevas versiones de PHP.

La Base de Datos necesita ser remodelada dando cabida a las funcionalidades de nuestros tiempos.

2.3 Perfil del Cliente - Usuario

El servicio Web clasifica los usuarios según sus funciones y sus privilegios entorno a la aplicación. Estos usuarios se dividen en 3 tipos: Administrador, Trabajadores y Usuarios.

2.3.1 Administrador

Es la persona que posee el control total sobre la aplicación.

Se encarga de las migraciones, actualizaciones, modificaciones y desarrollo de nuevas funcionalidades y/o módulos.

2.3.2 Trabajadores

Los trabajadores son las personas que se encargan de gestionar las incidencias de los usuarios finales. Tienen un menú de administración con privilegios limitados.

Dicho menú se usa para la gestión de cualquier incidencia, usuario y plaza así como la generación de estadísticas.

2.3.3 Usuarios

Son los usuarios finales, no tienen control sobre la administración del servicio Web.

Las funciones de estos usuarios están limitadas a la gestión de su usuario y sus plazas.

2.4 Objetivos

Para crear una aplicación sólida, manejable, cómoda, estructurada y de nuestros días se definen a continuación una serie de objetivos a cumplir.

- Aplicar el patrón de diseño MVC (Modelo – Vista – Controlador) para separar las capas y así conseguir mayor compatibilidad con la plataforma ya existente.
- Separar niveles de usuarios según los privilegios.
 - Nivel usuario: Gestionar usuarios y plazas de aparcamiento (alta, baja y modificación)
 - Nivel administrador: Gestionar usuarios, plazas, soporte para errores y estadísticas del servicio.
- Buscar plazas por calles.
- Mostrar la localización del parking en un mapa (Google maps).
- Mensajería interna.
- Encriptación de datos críticos.
- Alertar de cambios al correo ordinario.
- Posibilidad de adaptar el diseño para integrarlo en la Web de los clientes.
- Superar las pruebas del Test de Accesibilidad Web mínimo AA.
- Textos multi-idiomias.
- Libro de estilos.
- Enviar errores de funcionamiento.
- Enviar e-mails con recomendación del sitio Web a amigos.

2.5 Fuentes de Información

Para decidir en qué lenguaje y de qué manera se desarrolla el proyecto se consultan libros sobre lenguajes de programación tales como PHP, ASP, .NET y RoR así como contenidos online relacionados con el tema.

Para crear los contratos de venta o alquiler se ha consultado a una gestoría que nos proporciona los formularios y datos legales necesarios.

2.6 Descripción

En caso de disponer de una plaza de aparcamiento y durante unos días o una franja horaria no utilizarla tendrá la opción de alquilarla, intercambiarla e incluso venderla.

El usuario se da de alta en el servicio Web rellenando de manera correcta y con los datos verídicos los campos requeridos.

Una vez creada la ficha de usuario dispone de varias opciones como crear o buscar una plaza de aparcamiento.

En el caso de crear una nueva plaza tiene que introducir el tipo (alquiler, venta y/o intercambio), los datos de la localización, horario de disponibilidad y otras indicaciones básicas.

Si por lo contrario no dispone de aparcamiento pero quiere buscar uno simplemente debe indicar la localización dónde la quiere, la franja horaria en que necesita y el tipo que desea encontrar (alquiler, venta o intercambio).

Una vez encontrada tanto el propietario como el interesado se ponen en contacto mediante mensajería interna, acordando los precios y las condiciones.

Una vez ambos interesados están de acuerdo marcan la plaza como alquilada, vendida o intercambiada y descargan el fichero pdf con el contrato legal.

Acuerdan un día y en el momento de entrega de llaves del garaje se firma el contrato dando por finalizado el proceso.

Compartir SL en ningún momento obtendrá beneficios de las transacciones de sus usuarios.

2.7 Definición de etapas y metodología de desarrollo

Aquí se explican las fases que sigue el desarrollo de la aplicación.

1. Obtener una lista de problemas, deficiencias y mejoras para implantar en el sistema actual
2. Planificar y realizar entrevistas a diferentes niveles:
 - Jefes: dan una visión global del trabajo a realizar
 - Trabajadores: dan una visión detallada de las deficiencias del antiguo sistema. Proponen mejoras en función de las demandas cotidianas de los clientes
3. Ampliar la lista de problemas y deficiencias con las informaciones extrapoladas de las entrevistas
4. Desarrollar el estudio de viabilidad
 - Objetivos
 - Perfiles de usuarios
 - Etapas y metodología de desarrollo
 - Modelo de desarrollo
 - Recursos necesarios
 - Análisis Coste-Beneficio
 - Presupuesto
 - Riesgos
 - Alternativas
 - Etapas de desarrollo
5. Desarrollar el estudio funcional
 - Diseño de la Base de datos
 - Definir requisitos funcionales
 - Definir requisitos no funcionales
 - Definir el modelo de arquitectura MVC
6. Realizar el proyecto
 - Desarrollo de la aplicación
 - Desarrollo del diseño
 - Pruebas
7. Elaborar documentación
 - Manuales de Administración
 - Manuales de Usuarios
 - Memoria

2.8 Modelo de desarrollo

El modelo de desarrollo que utilizaremos será el evolutivo.

Iremos presentando el proyecto en formato de pantallas al cliente.

De esta manera nos podrá indicar los cambios, mejoras y/o nuevas funcionalidades que debemos implantar en nuestro diseño e implementación del producto final.

No se utiliza el lineal ni el de prototipo porque la retroalimentación con el cliente es fundamental para la creación de un servicio Web competente, actualizado y que cubra las necesidades más punteras.

2.9 Recursos

Aquí se detallan todos los requisitos tanto software como hardware necesarios para el desarrollo, mantenimiento y funcionamiento correcto de la aplicación.

2.9.1 Software

- Administrador
 - Programa de desarrollo: Notepad++
 - Servidor Web y desarrollo: WAMP (Apache + PHP 5.0 + MYSQL)
 - Sistema Operativo: GNU / Linux / Windows XP o superior
 - Navegador: Mozilla Firefox / Internet Explorer
 - Gestor de PDF's: Adobe Reader
 - Diseño y retoque fotográfico: Photoshop / Gimp
 - Serena OpenProj
- Trabajador / Usuario
 - Sistema Operativo: GNU / Linux / Windows XP o superior
 - Navegador: Mozilla Firefox / Internet Explorer
 - Gestor de PDF's: Adobe Reader
- Servidor
 - Base de datos: Mysql
 - Sistema Operativo: GNU / Linux
 - Servidor Web: Apache
 - Soporte para PHP 5.0 y JavaScript
 - Servidor FTP y Servidor DNS
 - Direcciones de correo y Usuarios Web

- Documentación:
 - OpenOffice / Microsoft Office XP o superior
 - Adobe Reader
 - Serena OpenProj / Microsoft Project

2.9.2 Hardware

- Trabajador / Usuario / Administrador
 - Memoria RAM: 256MB (recomendado 512MB)
 - Procesador: Intel / AMD a 1000Mhz
 - Disco Duro: 7GB libres
 - Teclado
 - Ratón mecánico (recomendado ratón óptico)
 - Monitor: Super VGA (800x600) o superior
 - Tarjeta de red: Ethernet 10/100
- Servidor
 - Memoria RAM: 1GB (recomendado 2GB)
 - Procesador Intel / AMD 2000Mhz
 - Disco Duro: 20GB SATA a 7200rpm con 16MB de buffer
 - Teclado y ratón
 - Monitor: Super VGA (800x600) o superior

2.10 Análisis Coste – Beneficio

A continuación se analizan los costes de material (Tabla 1.1: Coste de Material) y personal (Tabla 1.2: Coste de personal) necesarios para el desarrollo correcto y sin retrasos.

2.10.1 Coste de material

En la Tabla 1.1: Coste de Material se muestran todos los costes derivados de la compra o alquiler de material necesario para el funcionamiento o elaboración del proyecto.

Tabla 1.1: Coste de Material

Recursos	Coste Total	Coste mensual
Servidor	2.000€	
Conexión internet 6Mb		50,00€
JavaScript	0,00€	
WAMP	0,00€	
Notepad ++	0,00€	
Firefox	0,00€	
Adobe Reader	0,00€	
Open Proj	0,00€	
Amortización Photoshop	$(150/24)*8 = 50,00€$	
Amortización Windows XP	$(300/24)*8 = 100,00€$	
Amortización Office XP	$(250/24)*8 = 83,33€$	
Amortización PC programador	$(700/24)*8 = 233,33€$	
Total	2.466,66 €	50,00 €

2.10.2 Coste de personal

En la Tabla 1.2: Coste de Personal se muestran los costes desglosados por horas de todas las personas que han trabajado en el desarrollo e implementación del servicio Web.

Dado que en este caso el analista y el programador son la misma persona se asigna un precio hora unificado a los dos tipos de empleados.

El precio hora del Analista - Programador es de 20€/hora.

Tabla 1.2: Coste de Personal

Tarea	Trabajo (h)	Coste
Estudio de viabilidad	6	120,00€
Análisis funcional	45	900,00€
Diseño gráfico	5	100,00€
Diseño y configuración de la Base de Datos	4	80,00€
Programación	200	2.000,00€
Pruebas locales	23	460,00€
Instalación en el servidor público	5	100,00€
Elaborar la documentación	25	500,00€
Total	315	6.260,00€

2.11 Presupuesto

El presupuesto final con todos los gastos se desglosa en esta sección, aquí se puede ver el precio final del desarrollo e implantación de la aplicación.

Detalle del presupuesto

- **Servidor dedicado SD Pyme 2.880 €*
Memoria RAM: 2 GB DDR2
Procesador: INTEL Xeon 3065 2.33GHz con doble núcleo
RAID 1: 2 Discos Duros de 320 GB SATA a 7200rpm con 16MB de buffer
Tráfico mensual: 150 GB.**
- **Instalación en el servidor 144 €*
Configurar Apache, PHP 5.0 y Mysql
Instalar y configurar el Sistema Web para compartir plazas de aparcamiento**
- **Sistema Web para compartir plazas de aparcamiento 11.449,152 €*
Gestión de usuarios y plazas.
Mantenimiento de datos de usuarios, plazas y georeferencias.
Consultas referentes a usuarios o plazas.
Envío de correos automatizados según criterios.
Automatización de búsquedas por proximidad.
Georeferenciación de plazas en un mapa.**

Generación de estadísticas.

Base de datos de clientes, usuarios, plazas y georeferencias.

Interacción con el servicio Compartir Coche y la Web del cliente.

Detalle del presupuesto 14.473,152 €*

**Precios sin IVA.*

Beneficio = 1,2; Margen de Negociación = 1,2; Desviación = 1,3

2.12 Evaluación de riesgos

Todos los proyectos nuevos tienen unos elementos o eventos que pueden hacer fracasar el desarrollo, en este apartado se comentan los posibles riesgos y se plantean soluciones viables.

2.12.1 Rechazo del software

Implicar a los trabajadores en el desarrollo es una buena manera de evitar el rechazo, ya que, ellos proporcionan ideas para desarrollar el servicio cómodo y atractivo.

En su puesta en funcionamiento activar una sección de sugerencias es vital para la gestión de propuestas especificadas por los usuarios.

Todas las sugerencias se estudiarán y evaluarán según su viabilidad.

2.12.2 Seguridad

Al ser un servicio online está expuesto a ataques por la red. Para paliar estos ataques se realiza una copia de seguridad cada 15 días y se actualiza constantemente la seguridad del servicio.

2.12.3 Fallo de Internet / No disponer de red

Hay usuarios que pueden no tener Internet o que por ciertos motivos no dispongan de ello en ese momento para solucionar esto se activa un servicio complementario vía telefónica para hacer frente a las necesidades de los usuarios.

2.12.4 Fallo del servidor externo (Propio de cada cliente)

Este servicio Web carga dinámicamente contenidos de las Web de los clientes finales para una mejor integración y que en caso que el cliente desee modificar su Web se vea reflejado automáticamente en nuestra aplicación.

Para evitar que la caída, error o mal funcionamiento del servidor del cliente afecte a nuestra aplicación se crean automáticamente plantillas semanales en el servidor interno para que en caso de no recibir respuesta del servidor externo poder mostrar la última versión correcta conocida.

2.12.5 Fallo del servidor interno (Propio de la empresa)

En caso de caída, error o mal funcionamiento del servidor se abrirá una incidencia lo antes posible a la empresa que nos suministra el servidor para el restablecimiento lo antes posible.

Pese a que esto reducirá nuestra capacidad operativa se puede continuar ofreciendo el servicio gracias al servicio complementario de telefonía.

2.12.6 Incompatibilidad entre navegadores

Son habituales las incompatibilidades o interpretaciones diferentes de los estándares entre los diferentes tipos de navegadores.

Para solucionarlo se separa y personaliza de código y estilos para los diferentes tipos de navegadores (Mozilla e Internet Explorer).

2.13 Alternativas

En esta sección se explican las ventajas e inconvenientes de los diferentes lenguajes en los cuales podemos desarrollar el proyecto así como la conclusión sobre que tecnología utilizaremos.

2.13.1 Active Server Pages (ASP)

Lenguaje creado por Microsoft.

Ventajas:

- Estándar de Microsoft

Inconvenientes:

- Servidor con Internet Information Server (ISS) más caro que uno con Apache
- Las aplicaciones que dan soporte a este lenguaje son de pago
- No sigue el paradigma Modelo Vista Controlador (MVC)

2.13.2 .NET

Lenguaje creado por Microsoft.

Ventajas:

- Estándar de Microsoft
- A partir de Windows XP 2º Edición todos llevan integrados .NET
- Orientado a objetos

Inconvenientes:

- Servidor con ISS más caro que uno con apache
- Las aplicaciones que dan soporte a este lenguaje son de pago
- No sigue el paradigma Modelo Vista Controlador (MVC)
- Aprendizaje largo

2.13.3 PHP Hypertext Pre-processor (PHP)

Creado originalmente por Rasmus Lerdof.

La implementación principal es producida ahora por The PHP Group.

Ventajas:

- Lenguaje de programación interpretado
- Diseñado originalmente para la creación de páginas Web dinámicas
- PHP License, la Free Software Foundation considera esta licencia como software libre
- Mucha información y ejemplos de código
- Aprendizaje en corto periodo
- Última versión orientada a Objetos
- Es soportado por apache
- Metaprogramación
- Cómodo y robusto

Inconvenientes:

- No sigue el paradigma Modelo Vista Controlador (MVC)

2.13.4 Ruby on Rails (RoR)

Desarrollado por David Heinemeier Hansson.

Ventajas:

- Código abierto
- Sigue el paradigma de la arquitectura Modelo Vista Controlador (MVC)
- Orientado a Objetos
- Metaprogramación

- Cómodo y robusto

Inconvenientes:

- Versión estable pero se conoce que próximamente saldrá una nueva versión con cambios importantes
- Información menos extendida que PHP o ASP

2.13.5 Conclusión de las alternativas

Para el proyecto el que más se adapta al método de desarrollo es RoR pero pese que, en nuestro caso, es uno de los mejores se descartará por varias razones:

1. Porque el servicio Web tiene que ser compatible con la aplicación ya existente programada en PHP
2. Los cambios que se prevén en la nueva versión nos obligaría a rehacer parte del trabajo ya desarrollado y nos retrasaría en los plazos de entrega
3. El coste del servidor RoR es más alto que el de PHP
4. El desarrollador tiene que aprender este lenguaje y supondría costes y tiempos de formación

La empresa para la que se desarrolla el servicio Web es partidaria del software libre por ese motivo se descartan tanto ASP como .NET que pese a ser libres los software que dan soporte a estos lenguajes son propietarios.

Por todos estos motivos se descartará RoR, ASP y .NET y se desarrollará en el segundo lenguaje que se adapta a nuestro proyecto que es PHP.

2.14 Planificación

En la imagen 1.1 se desglosan todas las tareas del proyecto por horas necesarias para su desarrollo, también se indican las fechas de inicio y final de cada tarea así como las tareas predecesoras.

Se ha supuesto que la jornada laboral es de 4 horas por día, 20 horas al mes y que se trabajan 20 días al mes.

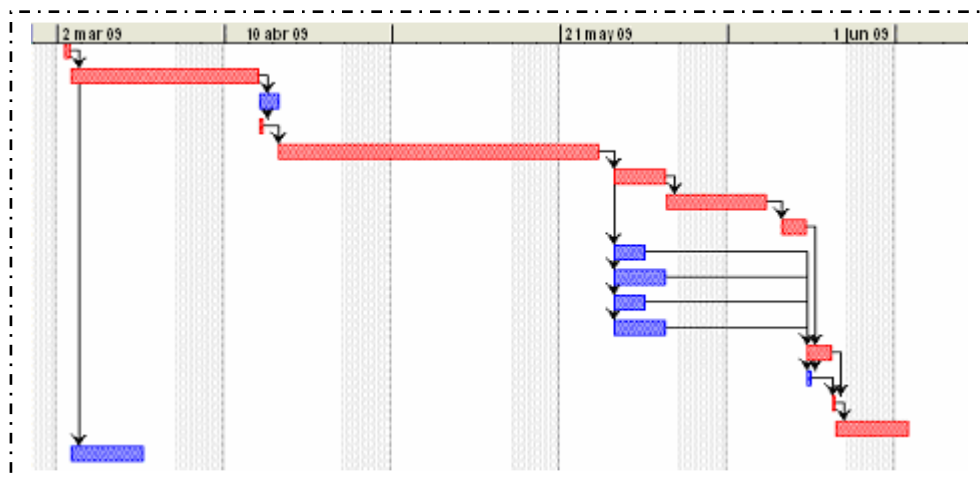
En la imagen 1.2 se presenta el diagrama de Gantt del proyecto utilizando Serena OpenProj.

Por último en la imagen 1.3 se amplía el diagrama de Gantt facilitando toda la información de horas necesarias y fechas previstas de inicio y finalización de cada tarea.

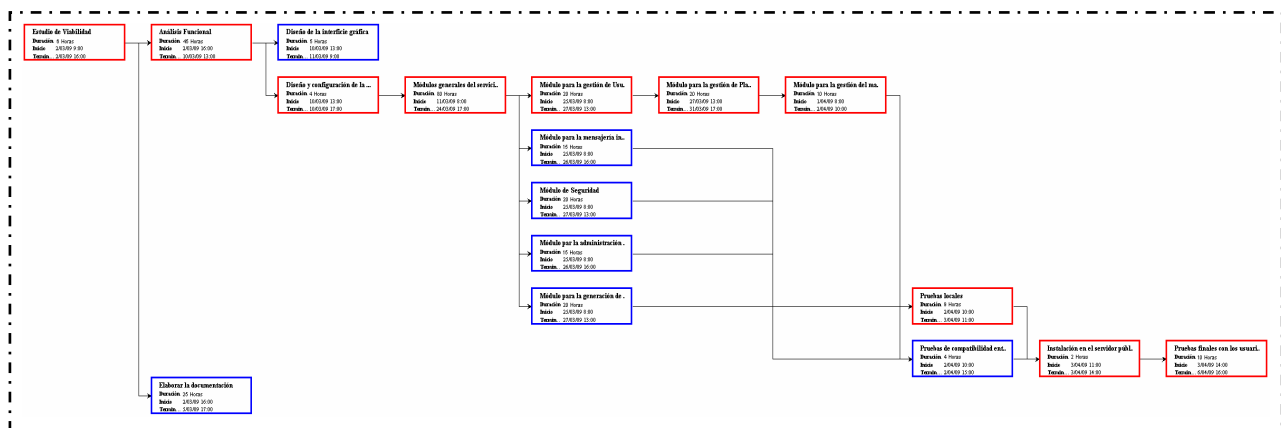
Capítulo 2: Planificación y desarrollo del proyecto

	Nombre	Duración	Inicio	Terminado	Predecesores
1	Estudio de Viabilidad	6 Horas	2/03/09 9:00	2/03/09 16:00	
2	Análisis Funcional	45 Horas	2/03/09 16:00	10/03/09 13:00	1
3	Diseño de la interficie gráfica	5 Horas	10/03/09 13:00	11/03/09 9:00	2
4	Diseño y configuración de la Base de Datos	4 Horas	10/03/09 13:00	10/03/09 17:00	2
5	Módulos generales del servicio Web	80 Horas	11/03/09 8:00	24/03/09 17:00	4
6	Módulo para la gestión de Usuarios	20 Horas	25/03/09 8:00	27/03/09 13:00	5
7	Módulo para la gestión de Plazas de aparcamiento	20 Horas	27/03/09 13:00	31/03/09 17:00	6
8	Módulo para la gestión del mapa	10 Horas	1/04/09 8:00	2/04/09 10:00	7
9	Módulo para la mensajería interna	15 Horas	25/03/09 8:00	26/03/09 16:00	5
10	Módulo de Seguridad	20 Horas	25/03/09 8:00	27/03/09 13:00	5
11	Módulo par la administración de la aplicación	15 Horas	25/03/09 8:00	26/03/09 16:00	5
12	Módulo para la generación de estadísticas	20 Horas	25/03/09 8:00	27/03/09 13:00	5
13	Pruebas locales	9 Horas	2/04/09 10:00	3/04/09 11:00	8;9;10;11;12
14	Pruebas de compatibilidad entre navegadores	4 Horas	2/04/09 10:00	2/04/09 15:00	8;9;10;11;12
15	Instalación en el servidor público	2 Horas	3/04/09 11:00	3/04/09 14:00	13;14
16	Pruebas finales con los usuarios y corrección de errores	10 Horas	3/04/09 14:00	6/04/09 16:00	15
17	Elaborar la documentación	25 Horas	2/03/09 16:00	5/03/09 17:00	1

(Imagen 1.1: Desglose de las fases)



(Imagen 1.2: Diagrama de Gantt)



(Imagen 1.3: Detalle del diagrama de Gantt)

2.15 Viabilidad legal

Con el propósito de cumplir la legislación se implantarán una serie de normas, derechos y medidas de seguridad.

1. Se comunicará al Registro Mercantil el nombre del dominio
2. En el servicio Web se incluye un apartado con el NIF, la denominación social, domicilio, correo electrónico, teléfono así como los datos de inscripción registral
3. Los datos personales que se requieren por la aplicación son de nivel básico
4. En caso de baja de un usuario se eliminarán los datos más críticos como nombres, apellidos, contraseña, correo electrónico, fecha de nacimiento, teléfonos y comentarios; dejando solamente información referente a las estadísticas
5. El servicio está dotado de herramientas al alcance de los usuarios para que éstos sean capaces de mantener sus datos actualizados
6. Tanto en la Web principal como en la de cada cliente se enlazarán un hipervínculo a un documento, explicando:
 - La existencia, tratamiento, finalidad y destinatario de los datos
 - El carácter obligatorio o voluntario de las respuestas
 - Consecuencias de la obtención de los datos o negación a suministrarlos
 - Derechos de acceso, rectificación, cancelación y oposición
 - Identidad y dirección del responsable de la Base de Datos
7. El usuario tiene derecho a:
 - Consulta al Registro Mercantil con la finalidad de obtener información de Compartir SL
 - Acceso periódico a sus datos
 - Rectificación y cancelación de sus datos
 - Oposición al tratamiento de sus datos
8. Compartir SL se compromete a:
 - No utilizar los datos con otros fines no especificados en el contrato
 - Todos los trabajadores de Compartir SL se comprometen a no divulgar los datos ni parcial ni totalmente
 - No comunicar los datos personales a terceros salvo lo autorice la ley, los datos recogidos sean de fuentes públicas o haya una relación entre este usuario y otro afectado
 - Mantener la integridad, disponibilidad y confidencialidad de los datos

- Crear un Log con el control de accesos y errores
- Sistema de identificación y autenticación
- Encriptación de las contraseñas
- Registro de incidencias
- Backups cada noche
- Gestión de soporte

Una vez puesto en funcionamiento los derechos de explotación son propiedad de la empresa Compartir SL.

2.16 Conclusiones

Por todo lo estudiado anteriormente, podemos concluir que crear el proyecto brinda unos beneficios funcionales y técnicos, así como permite mayor flexibilidad y disminución del tiempo de búsqueda de aparcamiento. Amplia el rango de búsqueda listándolo por factor de proximidad y lo presenta visualmente mediante un mapa.

La gestión de los datos de usuarios y plazas de aparcamiento se muestra ordenada y simplificada haciendo así que usuarios con pocos conocimientos informáticos puedan disfrutar de estos servicios. Como contra, tenemos todos los problemas típicos del desarrollo, tales como incompatibilidades entre navegadores o seguridad entre otros.

Sopesando todos estos motivos y los expuestos durante este estudio afirmamos que el proyecto resulta viable.

Capítulo 3

Requerimientos funcionales y no funcionales

3.1 Estudio de la Base de Datos

En este apartado se muestran las relaciones y los campos que pertenecen a las tablas de la Base de Datos así como información detallada de cada campo de las tablas.

Información	Nomenclatura
Clave Primaria (Primary Key)	Azul y Negrita
Clave Foránea (Foreign Key)	<u>Verde y Subrayado</u>
El campo puede ser nulo *	<i>Naranja y Cursiva</i>

* En todo el resto de casos el campo nunca será nulo.

** Los valores por defecto cuyo nombre sea Auto hacen referencia al tipo Auto Incremento.

3.1.1 Tabla de Usuario

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
id	Integer(11)	Auto**	Número identificador del usuario
nombre	Varchar(100)		Nombre del usuario
apellido	Varchar(100)		Apellido del usuario
email	Varchar(50)		Correo electrónico de contacto
password	Varchar(100)		Contraseña de entrada al servicio
idiomas	Varchar(50)		Idiomas que sabe hablar el usuario
idiomaInterfaz	Small Int(2)		Idioma de la aplicación
<u>idPais</u>	Varchar(2)	SP	Código de País
ciudad	Varchar(255)		Nombre de la ciudad de residencia
fechaNacimiento	Year(4)		Año de nacimiento
<i>telefonoFijo</i>	Big Int(11)		Teléfono Fijo
<i>telefonoMovil</i>	Big Int(11)		Teléfono Móvil
<i>codigoPostal</i>	Integer(5)		Código Postal
codigoSexo	Bool		Código del tipo de sexo 0- hombre 1- mujer
fechaAlta	Date		Fecha de creación del usuario
<u>idCobranded</u>	Integer(11)		Número identificador de cliente
<i>comentario</i>	Text		Comentario
ultimoLogin	Date		Última vez que entró al servicio
numeroLogins	Integer(11)	0	Número de veces que se ha logeado

tipoUsuario	Small Integer(1)	2	Código que nos informa sobre los privilegios del usuario 0- Administrador 1- Trabajadores 2- Usuarios
<i>fechaBaja</i>	Date		Fecha en que el usuarios se da de baja del servicio
<i>motivoBaja</i>	Small Integer(2)	0	Motivo por el cual se da de baja 0- No hay baja 1- Baja estándar

3.1.2 Tabla de Plaza

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
id	Integer(11)	Auto**	Número identificador de la plaza
<u>idUsuario</u>	Integer(11)		Número identificador del usuario
<u>idPaís</u>	Varchar(2)	SP	Código del País
<u>idCiudad</u>	Integer(11)		Número identificador de la ciudad
direccion	Varchar(255)		Calle y número en la que está la plaza de aparcamiento
<i>codigoPostal</i>	Integer(5)		Código Postal de la ciudad donde esta la plaza
latitud	Double		Latitud de la localización de la plaza
longitud	Double		Longitud de la localización de la plaza
<i>foto</i>	Varchar(255)		Dirección URL de la foto de la plaza
codigoFranja	Varchar(20)		Códigos que indican la disponibilidad de las plazas 1- Mañanas 2- Tardes 3- Noches 4- Fines de semana 5- Horario variable En caso que haya más de uno se separan con una “,”

codigoTamanyoCoche	Small Integer(2)		Código que indica el tamaño del coche que cabe en la plaza 1- Grande 2- Mediano 3- Pequeño
codigoAccion	Varchar(20)		Código que indica el tipo de acción que se hace con la plaza 1- Alquilar 2- Vender 3- Intercambiar Se separan los códigos por “,”
<i>precioVenta</i>	Double		Precio de venta
<i>precioAlquiler</i>	Double		Precio del alquiler por mes
<i>idUsuarioIntercambio</i>	Integer(11)		Número identificador del usuario con el que se intercambian las plazas
<i>idPlazaIntercambio</i>	Integer(11)		Número identificador de la plaza que se quiere intercambiar
<i>comentario</i>	text		Comentario sobre la plaza
estado	Small Integer(2)	1	Código que indica la disponibilidad de la plaza 1- Disponible 2- Intercambiada/Vendida 3- Cancelada
fechaAlta	Date		Fecha en la cual se da de alta una plaza
<i>fechaRenovacion</i>	Date		Fecha en que se renueva una plaza caducada

3.1.3 Tabla de historico plaza

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
id	Integer(11)	Auto**	Número identificador de la plaza
<u>idUsuario</u>	Integer(11)		Número identificador del usuario
<u>idPais</u>	Varchar(2)	SP	Código del País
<u>idCiudad</u>	Integer(11)		Número identificador de la ciudad
direccion	Varchar(255)		Calle y número en la que está la

			plaza de aparcamiento
<i>codigoPostal</i>	Integer(5)		Código Postal de la ciudad donde esta la plaza
latitud	Double		Latitud de la localización de la plaza
longitud	Double		Longitud de la localización de la plaza
<i>foto</i>	Varchar(255)		Dirección URL de la foto de la plaza
codigoFranja	Varchar(20)		Códigos que indican la disponibilidad de las plazas 1- Mañanas 2- Tardes 3- Noches 4- Fines de semana 5- Horario variable En caso que haya más de uno se separan con una “,”
codigoTamanyoCoche	Small Integer(2)		Código que indica el tamaño del coche que cabe en la plaza 1- Grande 2- Mediano 3- Pequeño
codigoAccion	Varchar(20)		Código que indica el tipo de acción que se hace con la plaza 1- Alquilar 2- Vender 3- Intercambiar Se separan los códigos por “,”
<i>precioVenta</i>	Double		Precio de venta
<i>precioAlquiler</i>	Double		Precio del alquiler por mes
<i>idUsuarioIntercambio</i>	Integer(11)		Número identificador del usuario con el que se intercambian las plazas
<i>idPlazaIntercambio</i>	Integer(11)		Número identificador de la plaza que se quiere intercambiar
<i>comentario</i>	text		Comentario sobre la plaza
estado	Small	1	Código que indica la disponibilidad

	Integer(2)		de la plaza 1- Disponible 2- Intercambiada / Vendida 3- Cancelada
fechaAlta	Date		Fecha en la cual se da de alta una plaza
<i>fechaRenovacion</i>	Date		Fecha en que se renueva una plaza caducada
fechaBaja	Date		Fecha en la que se ha dado de baja la plaza

3.1.4 Tabla de Ciudades

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
id	Integer(11)	Auto**	Número identificador de la ciudad
<u>idPais</u>	Varchar(2)	SP	Código de País
ciudad	Varchar(255)		Nombre de la ciudad
zona	Varchar(2)		Franja del globo terráqueo en la que se encuentra la ciudad
latitud	Double		Latitud de la ciudad
longitud	Double		Longitud de la ciudad
paisZona	Varchar(4)		Código que agrupa el país y la zona de la ciudad

3.1.5 Tabla de infoCobranded

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
id	Integer(11)	Auto**	Número identificador del cliente
nombre	Varchar(10)		Nombre del cliente
carpeta	Varchar(10)		Nombre de la carpeta en el servidor
<u>idPais</u>	Varchar(2)	SP	Código de País
urlCompartir	Varchar(255)		Dirección de la página hecha para el cliente
<i>urlCobranded</i>	Varchar(255)		Dirección de la página externa del cliente

coordenadas	Varchar(30)		Coordenadas geográficas se separa la latitud y longitud por “;”
idCiudad	Varchar(255)		Número identificador de la ciudad, en caso que requieran más de una ciudad se separan los números identificadores de las ciudades con “;”
idiomaDefecto	Small Integer(2)	2	Idioma por defecto de la página del cliente
idiomasDisponibles	Varchar(12)		Todos los idiomas disponibles en la página del cliente, si hay más de uno se separan por “;”
activo	Bool	0	Código que indica el estado del servicio 0- Aún no se ha implantado 1- Ya está en funcionamiento
tipoEntidad	Small Integer(2)	1	Define el tipo de entidad 1- Ciudad 2- Empresa 3- Mancomunidad 4- Consejo comarcal
estadisticas	Bool	0	Código que señala si se harán estadísticas o no. 0- Sin estadísticas 1- Con estadísticas

3.1.6 Tabla de Visitas

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
id	Integer(11)	Auto**	Número identificador de la visita
fecha	Date		Fecha actual
idCobranded	Integer(11)		Número identificador de Cliente
visitas	Integer(11)	0	Visitas en esa fecha
paginasVistas	Integer(11)	0	Páginas vistas en esa fecha

3.1.7 Tabla de TraducciónTextos

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
id	Integer(11)	Auto**	Número identificador del texto
codigo	Varchar(50)		Palabra que nos referencia con el texto
texto	text		Texto
idioma	Small Integer(2)		Idioma del texto 1- Catalán 2- Castellano 3- Inglés 4- Francés 5- Alemán 6- Italiano 7- Portugués 8- Vasco 9- Valenciano 10- Gallego

3.1.8 Tabla de Mensaje

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
id	Integer(11)	Auto**	Número identificador del mensaje
<u>idUsuarioEnvia</u>	Varchar(11)		Número identificador del usuario que envía
<u>idUsuarioRecibe</u>	Varchar(11)		Número identificador del usuario que recibe el mensaje
fecha	DateTime		Fecha de envío del mensaje
texto	Text		Mensaje
<u>idPlaza</u>	Integer(11)		Número identificador de la plaza sobre la que se va a hacer referencia en el mensaje
leido	Bool	0	Indica si un mensaje ha sido o no leído 0- No leído 1- Leído

<i>fechaLeido</i>	Date		Fecha en la que se ha leído el mensaje
-------------------	------	--	--

3.1.9 Tabla historicomensaje

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
id	Integer(11)	Auto**	Número identificador del mensaje
<u>idUsuarioEnvia</u>	Varchar(11)		Número identificador del usuario que envía
<u>idUsuarioRecibe</u>	Varchar(11)		Número identificador del usuario que recibe el mensaje
fecha	Date		Fecha de envío del mensaje
texto	Text		Mensaje
<u>idPlaza</u>	Integer(11)		Número identificador de la plaza sobre la que se va a hacer referencia en el mensaje

3.1.10 Tabla RecuperarContraseña

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
id	Integer(11)		Número identificador del recuperar contraseña
<u>idUsuario</u>	Integer(11)		Número identificador del usuario
hash	Varchar(100)		Hash de seguridad
horaCaducidad	DateTime		Día y hora en la que caduca el Hash

3.1.11 Tabla Alerta

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
id	Integer	Auto**	Número identificador de la alerta
<u>idUsuario</u>	Integer		Código de identificación del usuario
tipoAlerta	Small Int(2)		Código que identifica un tipo determinado de alerta 1- Mensajes sin leer

			2- Datos no actualizados
mensaje	Varchar(255)		Mensaje a mostrar al usuario
estado	Bool	0	Indica si el usuario ha visto o no la alerta 0- Vista 1- No vista

3.1.12 Tabla Paises

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
<u>idPais</u>	Varchar(2)		Código identificador para cada país
pais	Varchar(50)		Nombre del país
idioma	Small Int(2)	0	Indica el idioma en el que se encuentra el nombre del país 1- Catalán 2- Castellano 3- Inglés 4- Francés 5- Alemán 6- Italiano 7- Portugués 8- Vasco 9- Valenciano 10- Gallego

3.1.13 Tabla Zonas

Nombre del Campo	Tipo de Dato	Valor Defecto	Descripción
<u>paisZona</u>	Varchar(4)		Código que identifica cada agrupación de país y zona con la provincia en la que está.
provincia	Varchar(255)		Nombre de la provincia

3.2 Requerimientos funcionales

A continuación se exponen una serie de funcionalidades.

3.2.1 Módulo para la gestión de Usuarios

Funciones necesarias para la gestión y administración de los usuarios.

a) Dar de alta usuarios

Una vez rellenos los campos obligatorios crea un nuevo usuario en la aplicación.

Para poder disfrutar de todas las funcionalidades del servicio Web es necesario estar dado de alta como usuario.

b) Dar de baja usuarios

En caso que algún usuario decida dejar de utilizar el servicio puede usar la opción de darse de baja, la cual eliminará todos los datos críticos del usuario.

Delante del correo electrónico se pondrá NULL para indicar que está dado de baja. No se borra el usuario para tener constancia de este hecho en las estadísticas.

c) Modificar usuarios

Modifica los datos insertados anteriormente por el usuario.

d) Ficha de usuario

Muestra información no confidencial y pública sobre el usuario.

e) Enviar contraseña olvidada

Automáticamente la aplicación envía un mensaje al correo ordinario con un enlace válido durante 24 horas, en el cual se permite el cambio de contraseña.

Pasado este tiempo se anula la petición de cambio y se tiene que volver a solicitar en caso necesario.

f) Mostrar información sobre las plazas que tiene un usuario

Muestra la información referente a las plazas de aparcamiento que tiene un usuario así como el mapa de la localización de cada plaza.

En caso de tener un número elevado de plazas se utilizará la función de paginación (Apartado 3.3.1 a) Paginación) para la división en páginas.

g) Especificar motivo de una baja de un usuario

Este apartado se deja como ampliación por si algún día fuera necesario, en el proyecto se especifica siempre el motivo de baja 1 que es baja estándar.

3.2.2 Módulo para la gestión de Plazas de aparcamiento

Aquí se explican las funciones necesarias para tratar todas las acciones que requiere la gestión de plazas de aparcamiento.

a) Dar de alta plazas de aparcamiento

Después de rellenar todos los datos correctamente y georeferenciar la localización se crea una nueva plaza de aparcamiento que en las próximas búsquedas se tendrá en cuenta.

b) Modificar plaza de parking

Modifica los datos añadidos anteriormente referentes a la plaza de aparcamiento.

c) Mostrar información sobre las plazas de aparcamiento

Muestra la información sobre la plaza de aparcamiento así como la localización geográfica georeferenciada en google maps.

d) Listar todas las plazas de aparcamiento según un criterio

Se encarga de buscar las plazas de aparcamiento según los criterios especificados por el usuario. Crea un listado de todas las plazas ordenadas por grado de proximidad con el lugar de destino.

e) Cambiar el estado de la plaza de parking

En caso que un anuncio de plaza de aparcamiento sea comprado/alquilado por algún usuario permitirá cambiar el estado a no disponible o en caso de ya no disponer de la plaza poder anular el anuncio.

3.2.3 Módulo para la gestión del mapa

Se encarga del correcto funcionamiento de cualquier función que tenga que ver con el mapa.

a) Buscar por ciudad, calle y número

Busca en google maps las coordenadas con los datos introducidos posicionando el resultado en un mapa. Además mientras se escribe la ciudad y/o la calle se muestra un desplegable con un listado de las coincidencias hasta el momento.

b) Guardar la posición de la plaza de parking

Ya sea por una búsqueda o clicando directamente sobre el mapa guarda las coordenadas geográficas de la localización de la plaza.

3.2.4 Módulo para la mensajería interna

Módulo encargado de la gestión de correos internos entre los diferentes usuarios.

a) Enviar mensajes

La aplicación consta de mensajería interna que favorece a la protección de datos críticos como el correo electrónico. Cuando un usuario envía un e-mail se almacena una copia en la cuenta del usuario y se envía un mensaje al correo ordinario informando que tiene un correo en el Servicio Web para Compartir Plazas de Parking.

b) Listar mensajes

La función listar mensajes nos muestra ordenados por fecha de más recientes a más antiguos los mensajes que se han recibido de otros usuarios. En caso que el número de mensajes sea muy extenso se utilizará la función paginación (Apartad 3.3.1 a) Paginación) para listarlos correctamente en varias páginas.

c) Borrar Mensajes

Una vez leído el mensaje existe la posibilidad de borrarlo.

Los mensajes son guardados 1 vez y lo puede ver tanto el receptor como el emisor, por este motivo si uno de los dos desea eliminar el mensaje se coloca el valor NULL delante del id del usuario que lo ha querido borrado.

Así éste usuario no lo verá pero el otro sí.

Un script que se ejecuta cada día mediante la función Chron de Apache revisa la tabla de mensajes. Dicho script busca los mensajes que tanto emisor como receptor han borrado y los elimina de la tabla principal para copiarlos a la tabla de históricos.

d) Mostrar información acerca del Usuario que envía el correo

Muestra un enlace en el que está depositada la información pública del usuario que se comunica.

3.2.5 Módulo para la generación de estadísticas

Crea y gestiona documentos con varios tipos de datos estadísticos.

Para la creación de los gráficos se ha utilizado la librería de google chart.

a) Generación de gráficos verticales

Elabora un gráfico vertical con los datos que recibe.

b) Generación de gráficos horizontales

Crea un gráfico horizontal con los datos recibidos.

c) Generación de gráficos circulares

Muestra la información recibida en forma de gráfico circular.

d) Generación de las tablas de contenidos

Muestra en una tabla información estadística como la siguiente:

- Número de usuarios dados de alta en un mes por Cobranded
- Número de plazas creadas en un mes por Cobranded

3.2.6 Módulo para la administración de la aplicación

La administración de la aplicación consta de todas las funcionalidades de los Módulos funcionales:

- 3.2.1 Módulo para la gestión de Usuarios
- 3.2.2 Módulo para la gestión de Plazas de aparcamiento
- 3.2.3 Módulo para la gestión del mapa
- 3.2.5 Módulo de estadísticas

Y también de los Módulos no funcionales:

- 3.3.1 Módulos generales del servicio Web

- 3.3.2 Módulo para la gestión de Usuarios
- 3.3.3 Módulo para la gestión de Plazas de aparcamiento
- 3.3.4 Módulo para la gestión del mapa
- 3.3.6 Módulo de Seguridad

Así como algunas funciones nuevas como:

a) Alta de Cobrandeds / Alta de nuevos clientes

Se trata de añadir todos los datos referentes a los nuevos clientes y los parámetros de personalización que necesitan sus servicios Web.

b) Modificar Cobrandeds / Modificar clientes

Modifica los datos de los clientes o sus parámetros de personalización.

3.3 Requerimientos no funcionales

A continuación se especifica el siguiente listado de requerimientos no funcionales.

3.3.1 Módulos generales del servicio Web

En esta sección se explican detalladamente las características y utilidades de las funciones generales del programa.

a) Paginación

Cuando la información que se visualiza es demasiado abundante se divide en páginas.

Cada página contiene una serie de bloques de información.

Los índices que nos indican cómo movernos por estas páginas son la paginación.

En el servicio Web se utiliza en secciones, como la lista de plazas de aparcamiento o la lista de mensajes enviados/recibidos.

b) Enviar correos electrónicos

Posibilita el envío de correos electrónicos a correos ordinarios.

Una de sus aplicaciones es el envío de resultados interesantes a amigos o recomendar la página.

c) Cambiar la navegación

A medida que el usuario avanza por las diferentes secciones de la aplicación esta función hace que la navegación vaya cambiando.

d) Logearse

La función logearse recibe los datos desde un formulario y hace una búsqueda en la Base de Datos para averiguar si la persona que intenta entrar es usuaria de la aplicación o no. En caso afirmativo mostrará más opciones en el menú y en caso negativo nos informará que no somos usuarios y cómo nos podemos dar de alta.

e) Des-Logearse

Con esta función conseguimos finalizar la sesión de nuestro usuario.

Una vez finalizada, todos los datos son eliminados del ordenador quedando solamente constancia de los cambios guardados en el servicio Web.

Cualquier persona que posteriormente utilice la aplicación no tendrá acceso a ninguno de los datos utilizados en la sesión anterior.

f) Control de Sesiones

Se encarga de controlar, mantener y asegurar la integridad física de los datos en las sesiones.

g) Importar las páginas de los clientes finales

Esta función dota a la aplicación de métodos para la importación dinámica de las Web de los clientes, facilitando así, la adaptación y personalización del servicio.

h) Crear plantillas en nuestro servidor con las páginas de los clientes

Como es posible que las páginas de los clientes en algún momento fallen, periódicamente se hacen copias de éstas en el servidor de Compartir SL.

De esta manera aunque el servidor del cliente no funcione se puede continuar mostrando correctamente la aplicación.

i) Idioma

La Base de Datos tiene todos los textos en varios idiomas para poder traducir la aplicación en caso de que sea necesario.

3.3.2 Módulo para la gestión de Usuarios

Funciones necesarias para la gestión y administración de los usuarios.

a) Contar el número de veces que un usuario se ha conectado al Servicio

Cuenta el número de veces que un usuario entra en el servicio Web.

b) Última vez que ha entrado

Muestra la fecha de la última vez que el usuario se conectó a la aplicación.

3.3.3 Módulo para la gestión de Plazas de aparcamiento

Aquí se explican las funciones necesarias para tratar todas las acciones que requiere la gestión de plazas de aparcamiento.

a) Hacer caducar las plazas antiguas

Los anuncios de las plazas caducan cada X meses, dando la posibilidad al creador de renovar el anuncio otros X meses.

3.3.4 Módulo para la gestión del mapa

Se encarga del correcto funcionamiento de cualquier función que tenga que ver con el mapa.

a) Interactuar el ratón con el mapa

Dado que es posible que la precisión en alguna búsqueda no sea la idónea o el usuario no conozca la localización exacta de la plaza, existe la posibilidad de posicionar la localización de la plaza pinchando directamente sobre el mapa o arrastrando el marcador que indica la localización de la plaza con el ratón.

b) Mostrar localización en Mapa

Se encarga de georeferenciar las coordenadas con google maps.

3.3.5 Módulo para la mensajería interna

Módulo encargado de la gestión de correos internos entre los diferentes usuarios.

a) Control de mensajes sin leer

Muestra con un color diferente los mensajes que aún no han sido leídos facilitando así de manera visual la certeza de haber leído o no un correo.

3.3.6 Módulo de Seguridad

Se encarga de controlar la integridad de los datos así como proteger de posibles ataques al servicio Web.

a) Encriptación contraseña

Recoge la contraseña y la codifica utilizando md5, guarda el resultado en la Base de Datos.

b) Control de SQL injection

Función que filtra todos los envíos de información evitando inyecciones de SQL.

c) Control contra ataques XSS

Función que filtra los datos eliminando código malicioso.

c) Control de Cobrandeds

Se encarga de comprobar que el cliente existe y tiene activado el servicio, en caso afirmativo se muestra la Web y en caso negativo se redirige a un buscador de clientes.

d) Control de acciones

Esta función se encarga de verificar que la acción a la que se está accediendo existe y se tiene los permisos suficientes (Apartado 2.3 Perfil del Cliente - Usuario) para poder utilizarla.

En caso de no cumplirse cualquiera de los dos requisitos anteriores nos informará del error/restricción.

e) Control de secciones

Comprueba que existe la sección a la cual se intenta acceder, en caso de no existir redirigirá a la página principal.

Si después de la sección no se especifica la acción a realizar automáticamente realizará la acción que tenga cada sección marcada como acción por defecto.

f) Filtro de IP's

Dado que hay clientes que informan que solamente admiten el acceso a su servicio Web de ciertos ordenadores, se dota al Servicio Web de compartir plazas de aparcamiento con un filtro de IP's para la restricción, en caso de ser necesaria, de las IP's entrantes.

Dicho filtro es individual para cada cliente dando la posibilidad de crear una especie de Red privada virtual (VPN).

g) Control de envío de mensajes

Se controla que todos los mensajes que se envían mediante la mensajería interna vayan destinados a usuarios concretos y que estén dados de alta en ese momento.

h) Control de todos los campos

Además del filtro de SQL injection (3.1.6.2 *Control de SQL injection*) se filtran todos los datos que se envían para evitar el uso de caracteres prohibidos o códigos maliciosos.

3.3.7 Módulo Alertas

Es el módulo encargado de mostrar una serie de alertas a los usuarios.

a) Mostrar alerta

Al conectarse un usuario se encarga de mostrar automáticamente las novedades que tiene. Un ejemplo de novedad es la recepción de un correo electrónico.

b) Desactivar alerta

Cuando el usuario atiende la alerta queda desactivada, así al entrar la siguiente vez no le muestra el aviso de información nueva.

Capítulo 4

Interfície

4.1 Interficie de la aplicación

En la actualidad se está renovando el aspecto de la empresa, modernizando sus páginas a diseños más actuales. Este hecho conlleva actualizar todas las páginas de sus productos a una misma estética.

Con la intención de preservar esta estética el sistema para compartir plazas de aparcamiento seguirá el modelo y las normas del nuevo diseño.



(Imagen 1 Pantalla de inicio)

En la cabecera se han utilizado tonos marrones pastel que da una connotación a tierra y medio ambiente.

Tanto el azul de la P como el verde del logo de la empresa son resaltados ya que rompen con la monotoneidad de la cabecera.

Es importante resaltar estos dos elementos ya que con un simple vistazo nos hacemos una idea sobre la temática del servicio.

El menú en un tono marrón un tanto más oscuro para los elementos seleccionados nos ayuda a ver en que sección estamos sin romper la armonía de la Web así como la flecha que aparece debajo del ítem seleccionado.

Los enlaces para el cambio del idioma está también en tono marrón pastel pero al pasar por encima rápidamente nos llama la atención el naranja chillón que aparece, este contraste de color nos informa rápidamente que estamos sobre un elemento clicable.

Los títulos como “Compartir Parking” en negrita y más grandes nos informan de manera vistosa y clara de la sección en la que estamos.

Capítulo 4: Interficie

El marrón caqui de los subtítulos es llamativo pero a su vez conserva la dinámica y armonía del sitio.

La pantalla de “Entrar” es sencilla, clara y concisa.

No se han añadido elementos inútiles, simplemente se pide el correo electrónico y la contraseña del usuario. Debajo de botón Entrar un enlace acorde con los colores de la Web nos informa que cliquemos sobre el si necesitamos recuperar la contraseña.

Los campos obligatorios se marcan con rojo para que el usuario rápidamente se dé cuenta de su importancia.

(Imagen 2 Pantalla de entrar)

Por otro lado está la pantalla de “Registrarme”, dicha pantalla al igual que todo el resto de formularios sigue una línea muy básica.

Los nombres de los campos en color negro y posteriormente en la línea de abajo, en caso que sea necesario, un aviso rojizo advirtiendo que el campo es obligatorio e importante.

(Imagen 3 Pantalla de registrarme)

Capítulo 4: Interficie

En la pantalla “Nueva plaza” se ha incrustado el mapa de google maps a la derecha con la finalidad que el formulario no quedara tan pobre y ya de paso mostrar la información de la plaza en tiempo real.

Al igual que en la pantalla explicada anteriormente (*“Registrarme”*) los campos necesarios se han marcado con un color rojizo para que rápidamente sepamos que son importantes.

Los Radio Button y check Box en una misma línea hacen que la página no se extienda innecesariamente y cree sensación de demasiados datos a completar.

The screenshot shows the 'Nueva Plaza' (New Parking Space) form on the 'Pcompartir' website. The form is located on the left side of the page, and a Google Map of Spain is on the right side. The form fields are as follows:

- Ciudad:** Campo necesario (red text)
- Dirección:** Campo necesario (red text)
- Código Postal:**
- Fotografía de la plaza:** Examinar...
- Horario de disponibilidad de la plaza:** Campo necesario (red text). Options: ☐ Mañanas ☐ Tardes ☐ Noches ☐ Fines de semana ☐ Horario variable
- Tamaño del coche:** Campo necesario (red text). Options: ☐ Grande ☐ Mediano ☐ Pequeño
- ¿Qué quieres hacer con la plaza?:** Campo necesario (red text). Options: ☐ Alquilar ☐ Vender ☐ Intercambiar
- Precio de venta:**
- Precio de alquiler (€/mes):**
- Comentario:**
- crear plaza** (button)

The Google Map on the right shows the location of the parking space. The map is titled 'Actualitzar mapa' and 'Mapa'. The map shows the location of the parking space in Spain, with a red pin indicating the location. The map is powered by Google.

ca | es

Inicio Nueva Plaza Buscar Plaza Mis Plazas Mis Mensajes Mis datos Darse de baja Administración Salir

Compartir > Plaza > Nueva Plaza

Nueva Plaza

Ciudad
Campo necesario

Dirección
Campo necesario

Código Postal

Fotografía de la plaza
Examinar...

Horario de disponibilidad de la plaza
Campo necesario
☐ Mañanas ☐ Tardes ☐ Noches ☐ Fines de semana ☐ Horario variable

Tamaño del coche
Campo necesario
☐ Grande ☐ Mediano ☐ Pequeño

¿Qué quieres hacer con la plaza?
Campo necesario
☐ Alquilar ☐ Vender ☐ Intercambiar

Precio de venta

Precio de alquiler (€/mes)

Comentario

crear plaza

Preguntas frecuentes | Díselo a un amigo | Errores de funcionamiento | Accesibilidad | Política de privacidad | Condiciones de uso

© Compartir SL

(Imagen 4 Pantalla de nueva plaza)

Capítulo 4: Interficie

El buscador de plazas es sencillo y claro, sin necesidad de recargarlo con mapas o información innecesaria.



(Imagen.5 Pantalla de buscar plaza)

Las pantallas de ficha de usuario y detalle de la plaza son casi iguales, la única diferencia es que la de detalle de la plaza tiene la fotografía de la plaza y el mapa mientras que la ficha de usuario solamente tiene la información.

En ambos casos se utiliza la negrita para resaltar que tipo de información se está mostrando y texto sin negrita para mostrar esos datos.



(Imagen 6 Pantalla de ficha de la plaza)

Capítulo 4: Interficie

En caso de querer intercambiar se muestra un listado simple donde sólo nos deja seleccionar una de las plazas que tenemos activada para este motivo, en caso que no haya ninguna nos muestra un mensaje de error con el estilo explicado anteriormente. El título con letra en negrita y más grande es lo único que resalta así no hay ningún elemento que le quite importancia a la elección de la plaza.



(Imagen 7 Pantalla de intercambiar)

El listado de las plazas que un usuario tiene y el de mensajes enviados/recibidos sigue en la línea del resto de pantallas.

Un rectángulo ancho y largo con un borde más oscuro que el interior, ambos en los mismos tonos de marrón que la cabecera.

Las palabras claves en negrita y lo que ha seleccionado el usuario sin negrita hacen que sea fácilmente leíbles.

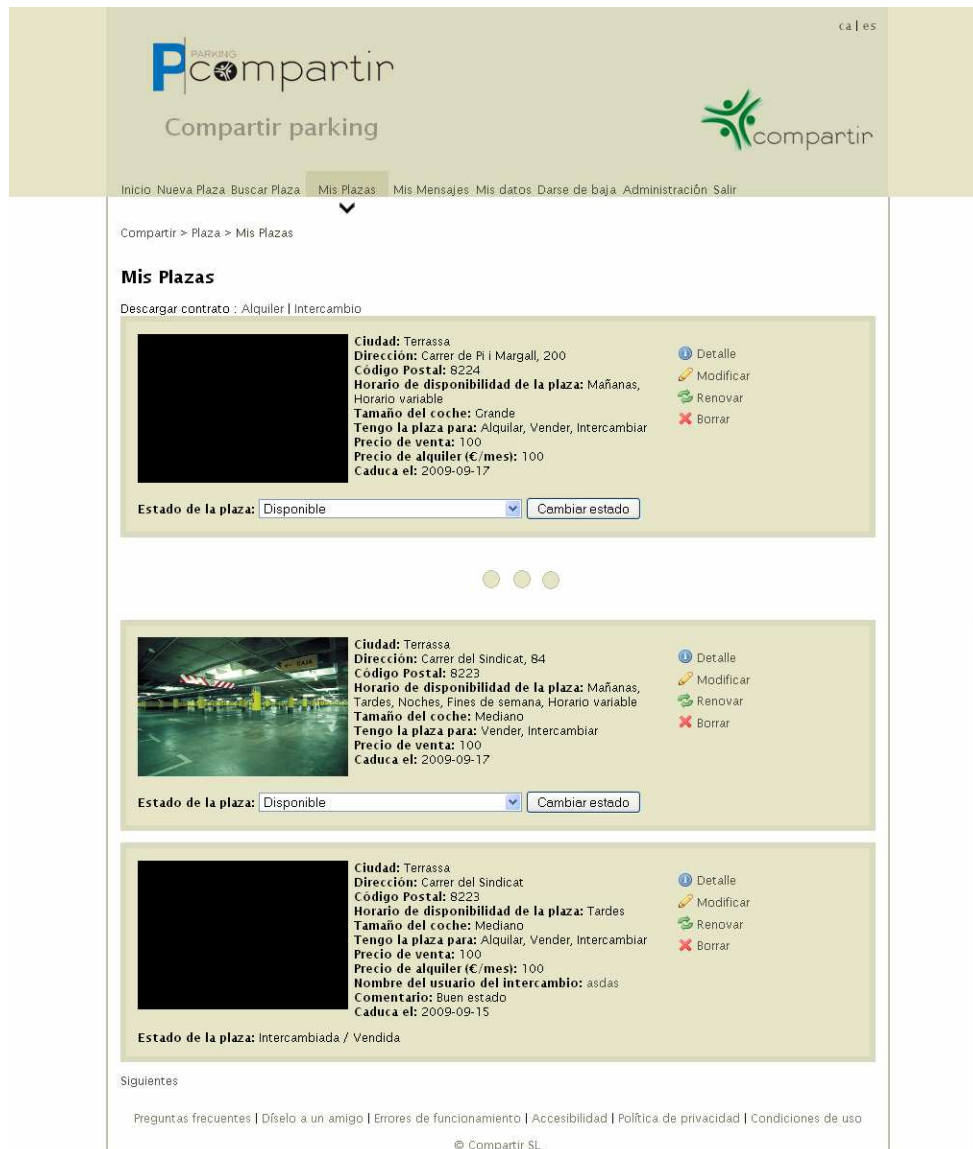
En el caso del listado de plazas hay también un menú lateral derecho con los enlaces relacionados de cada plaza, dichos enlaces siguen el mismo tono marronzoso de todos los hipervínculos pero una imagen delante del enlace hacen atractivo el listado.

Al separar la disponibilidad y ponerla debajo de la imagen da una sensación de amplitud y desglosación de la información según tipos.

En la parte superior un discreto enlace a descargar los contratos y en la parte final otro sencillo enlace para ir a la siguiente página del listado hacen que la página sea elegante y nada recargada.

En el caso del listado de mensajes enviados/recibidos consta de una cabecera con el mensaje hacia quien se envía/recibe así como la hora del envío/recepción.

El menú de la derecha consta de imágenes que resaltan las acciones que podemos realizar sobre nuestros mensajes.

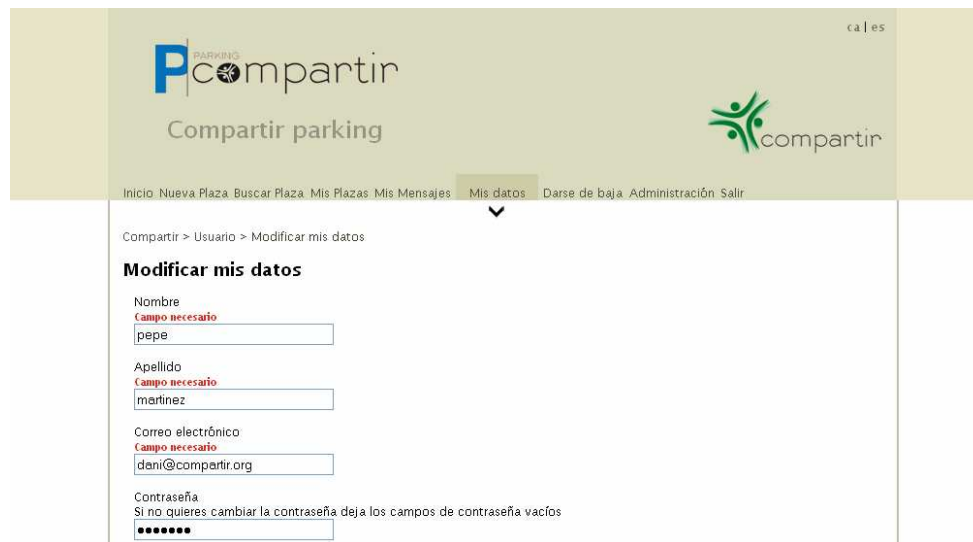


(Imagen 8 Pantalla de listado de plazas)

En caso de no haber plazas o mensajes se muestra una pantalla explicativa informando de tal hecho.

Capítulo 4: Interficie

La pantalla de “Mis datos” con total seguridad podemos decir es una de las más sobrias. Dado que sólo se muestran las cajas de texto y demás elementos que el usuario tiene que rellenar para cambiar sus datos. Los campos importantes nuevamente resaltados en rojo.



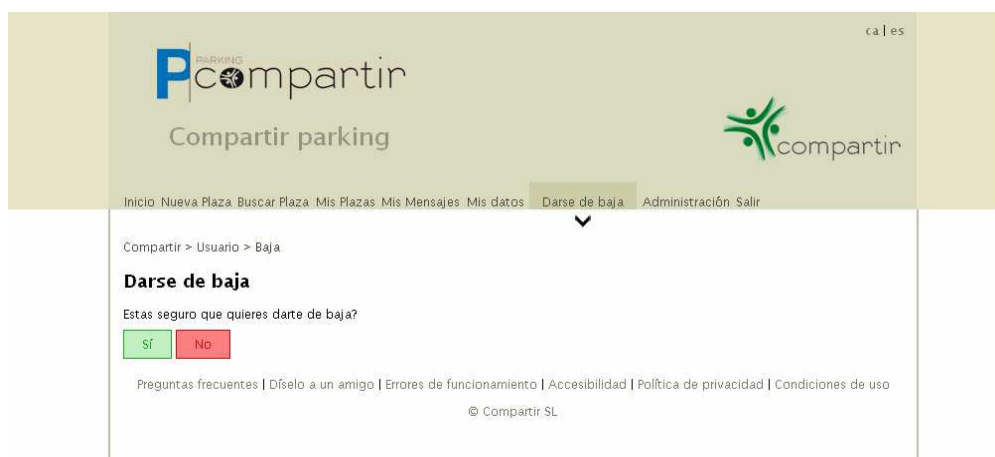
(Imagen 9 Pantalla de mis datos)

Todas las pantallas de baja/borrar, tanto de usuario, de plaza y/o mensaje muestran este mismo aspecto.

Un mensaje concreto y claro preguntando si se está seguro de lo que va a realizar y dos botones bien llamativos con “Sí” y “No”.

El “Sí” se ha puesto de color verde ya que este color se asocia a cosas positivas mientras que el “No” se ha puesto de color rojo que se asocia con cosas negativas.

Dado que borrar y dar de baja son destruir elementos se ha considerado apropiado este color.



(Imagen 10 Pantalla de darse de baja)

A continuación se muestra la interficie de la sección de administración.

En la pantalla de “Nuevo cliente” se siguen los mismos patrones que en el resto de formularios. Pantallas simples donde se remarcan los datos obligatorios.

Siempre en concordancia con el resto de estética de la Web.

The screenshot shows the 'Nuevo cliente' form within the 'Administración' section of the 'Compartir parking' website. The header includes the 'Pcompartir' logo and navigation links like 'Inicio', 'Nueva Plaza', 'Buscar Plaza', etc. The breadcrumb trail is 'Compartir > Administración > Nuevo cliente'. On the left, there's a sidebar with 'Nuevo cliente' (highlighted), 'Modificar cliente', and 'Estadísticas del cliente'. The main form area contains several input fields, each with a red 'Campo necesario' (Required field) label: 'Nombre', 'Nombre de la carpeta del cliente', 'Dirección URL de compartir', and 'Dirección Web del cliente'. A 'Ciudad' label is partially visible at the bottom.

(Imagen 11 Pantalla de nuevo cliente)

Tanto en la pantalla de “Modificar Cliente” como “Estadísticas del Cliente” se muestra un desplegable con todos los clientes dados de alta para que podamos seleccionar uno y realicemos la acción deseada.

The screenshot shows the 'Modificar cliente' form. The breadcrumb trail is 'Compartir > Administración > Modificar cliente'. The sidebar highlights 'Modificar cliente'. The main form area features a dropdown menu with 'Terrassa' selected and a 'Modificar' button. At the bottom, there are links for 'Preguntas frecuentes', 'Díselo a un amigo', 'Errores de funcionamiento', 'Accesibilidad', 'Política de privacidad', and 'Condiciones de uso', followed by the copyright notice '© Compartir SL'.

(Imagen 12 Pantalla de modificar cliente)

En el caso de “Modificar cliente” nos parece la siguiente pantalla, una vez más siguiendo la misma línea de los formularios anteriores.

ca | es

Compartir parking

Inicio Nueva Plaza Buscar Plaza Mis Plazas Mis Mensajes Mis datos Darse de baja Administración Salir

Compartir > Administración > Modificar cliente

Modificar cliente

Nuevo cliente

Modificar cliente

Estadísticas del cliente

Nombre
Campo necesario
Terrassa

Nombre de la carpeta del cliente
Campo necesario
terrassa

Dirección URL de compartir
Campo necesario
<http://terrassa.compartirp>

Dirección Web del cliente
Campo necesario
<http://terrassa.com>

(Imagen 13 Pantalla de modificar cliente)

Capítulo 4: Interficie

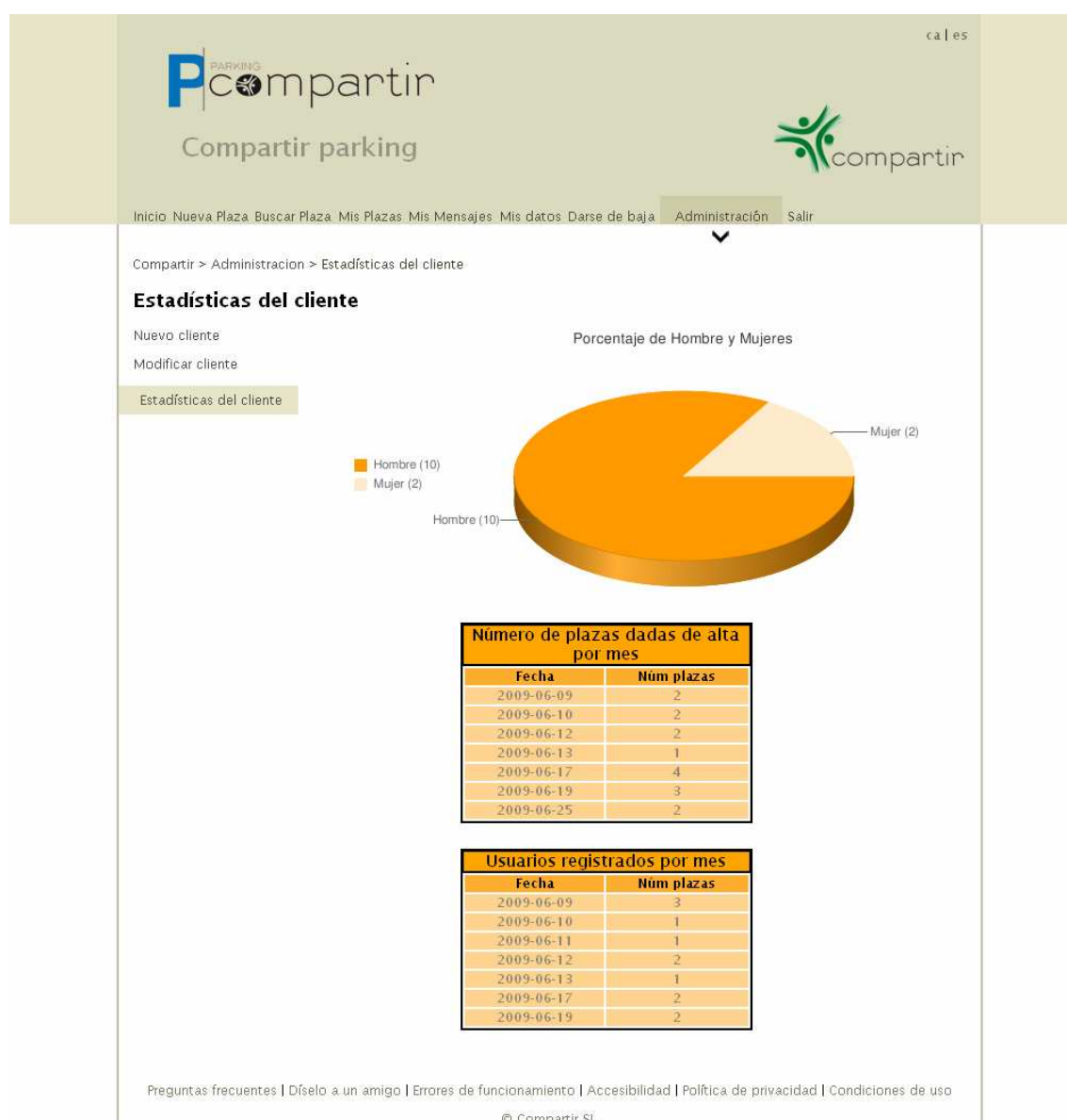
En el caso de “Estadísticas del cliente” se nos muestra un gráfico circular con el porcentaje de hombre y mujeres que utilizan el servicio para dicho cliente.

Además también se muestran unas parrillas con el número de plazas dadas de alta desglosadas por meses y número de usuarios también desglosados por meses.

Se utiliza el color naranja para llamar la atención y resaltar los datos más importantes.

Las tablas tienen un degradado de color naranja para denotar títulos, subtítulos y datos (de mayor a menor intensidad).

El color naranja impacta sobre los tonos pastel del servicio Web pero a su vez no desentona drásticamente.



(Imagen 14 Pantalla de estadísticas del cliente)

Capítulo 4: Interficie

Por otro lado tenemos las pantallas de “Preguntas frecuentes”, “Accesibilidad”, “Política de privacidad” y “Condiciones de uso”, todas estas pantallas sólo contienen información explicativa.

Los títulos se han puesto en negrita mientras que las explicaciones tienen un color negro.



(Imagen 15 Pantalla de preguntas frecuentes)

También están las pantallas de “Díselo a un amigo” y “Errores de funcionamiento”, ambas manteniendo el estilo de los formularios, nombre de los campos en negro y rojo para indicar campos obligatorios.

La pantalla “Errores de funcionamiento” a su vez tiene de color azul claro una breve explicación sobre que se debe introducir en uno de los campos.

Este azul resalta al estar justo debajo del rojo que indica campo importante.

Al ser un azul suave resalta pero queda en segundo lugar ya que el rojo es lo que más nos llama la atención.

The screenshot shows the 'Errores de funcionamiento' page. At the top, there's a navigation bar with the 'Pcompartir' logo and a 'ca | es' language selector. Below the navigation bar, there's a breadcrumb trail: 'Inicio Nueva Plaza Buscar Plaza Mis Plazas Mis Mensajes Mis datos Darse de baja Administración Salir'. The main content area is titled 'Compartir > Errores de funcionamiento'. It starts with a section '1. Describe el Problema' which includes a 'Campo necesario' label and a text input field. Below this, there's a description prompt: 'Describe, con tanto detalle como puedas el Problema.' followed by another 'Campo necesario' label and a large text area. A dropdown menu asks '¿Cuántas veces te ha pasado?' with the option 'Selecciona una opción'. The next section is '2. Información de contacto', which includes fields for 'Nombre' and 'Correo electrónico', both marked as 'Campo necesario', and an 'Enviar' button. At the bottom, there's a footer with links: 'Preguntas frecuentes | Díselo a un amigo | Errores de funcionamiento | Accesibilidad | Política de privacidad | Condiciones de uso' and a copyright notice '© Compartir SL'.

(Imagen 16 Pantalla de errores de funcionamiento)

La pantalla de “Has olvidado contraseña?” simplemente nos muestra un campo con el correo electrónico, es claro y sencillo, no necesita más explicación ni colorido.

The screenshot shows the 'Contraseña olvidada' page. It has the same header and navigation bar as the previous page. The breadcrumb trail is 'Inicio Nueva Plaza Buscar Plaza Registrarme Entrar'. The main content area is titled 'Compartir > Usuario > Contraseña olvidada'. It features a section 'Contraseña olvidada' with a 'Correo electrónico' label and a text input field. Below the input field is an 'Enviar' button. The footer is identical to the previous page, with links: 'Preguntas frecuentes | Díselo a un amigo | Errores de funcionamiento | Accesibilidad | Política de privacidad | Condiciones de uso' and a copyright notice '© Compartir SL'.

(Imagen 17 Pantalla de has olvidado contraseña)

Una vez recuperada la contraseña se nos envía un correo electrónico con una dirección para cambiar nuestra contraseña por otra. Dicha pantalla es la siguiente:



The screenshot shows the 'Recuperar contraseña' page. At the top, there's a header with the 'Pcompartir' logo and navigation links: 'Inicio', 'Nueva Plaza', 'Buscar Plaza', 'Registrarme', and 'Entrar'. Below the header, the breadcrumb trail reads 'Compartir > Usuario > Recuperar contraseña'. The main heading is 'Recuperar contraseña'. There are two input fields: 'Contraseña' and 'Repita la contraseña', followed by a 'Cambiar' button. At the bottom, there are links for 'Preguntas frecuentes', 'Díselo a un amigo', 'Errores de funcionamiento', 'Accesibilidad', 'Política de privacidad', and 'Condiciones de uso', along with a copyright notice '© Compartir SL'.

(Imagen 18 Pantalla de recuperar contraseña)

Podemos ver también su sencillez, dado que en el correo ya se explica para que sirve esta pantalla no hace falta recargarla con mensajes redundantes ni colores llamativos.

Para finalizar falta mostrar las pantallas de Alertas de error al introducir algún campo, de Información de actividad reciente y de Información estándar.

La pantalla de Alerta de error nos muestra un rectángulo rojo con el título "Alerta", se han evitado las palabras como "Error, Fail, Mal Funcionamiento" dado que tienen una connotación que para el usuario puede ser catastrófica y dar la impresión de mal funcionamiento del servicio.

El cuerpo del rectángulo es de un rojo menos intenso simulando así una ventana de windows. Al clicar sobre la alerta automáticamente nos lleva al campo que está equivocado para mayor comodidad.



The screenshot shows the 'Entrar' page. The header is identical to the previous page. The breadcrumb trail is 'Compartir > Usuario > Login'. A red alert box is displayed at the top, with the title 'Alerta' and the message 'Nombre de usuario o contraseña incorrecta'. Below the alert, the heading is 'Entrar'. There are two input fields: 'Correo electrónico' (with a red 'campo necesario' label) containing 'dani@compartir.org', and 'Contraseña' (with a red 'campo necesario' label) containing masked characters. An 'Entrar' button is at the bottom. A link 'Has olvidado la contraseña?' is also present.

(Imagen 19 Pantalla con un mensaje de alerta de error)

Capítulo 4: Interficie

La pantalla de Información de actividad reciente, es de color naranja.
Se ha seleccionado este color por el grado de alerta.
El rojo simboliza que alguna cosa está mal pero el naranja connota un aviso no importante.



(Imagen 20 Pantalla con un mensaje de información de actividad reciente)

Para acabar el mensaje de información estándar también en forma de rectángulo, este es de mismo tono marrón del servicio Web.

Dado que es información sobre que se debe hacer, como unos consejos a seguir no se ha considerado necesario utilizar colores con connotación subjetiva.

Para preservar la monotonía del colorido se ha optado por utilizar los mismos tonos marrones de la cabecera.

Al mismo tiempo el cuerpo del rectángulo es de color blanco para facilitar su lectura.

Si se clicca sobre el mensaje se redirige a la acción que se cree que es la que el usuario está interesado en hacer.



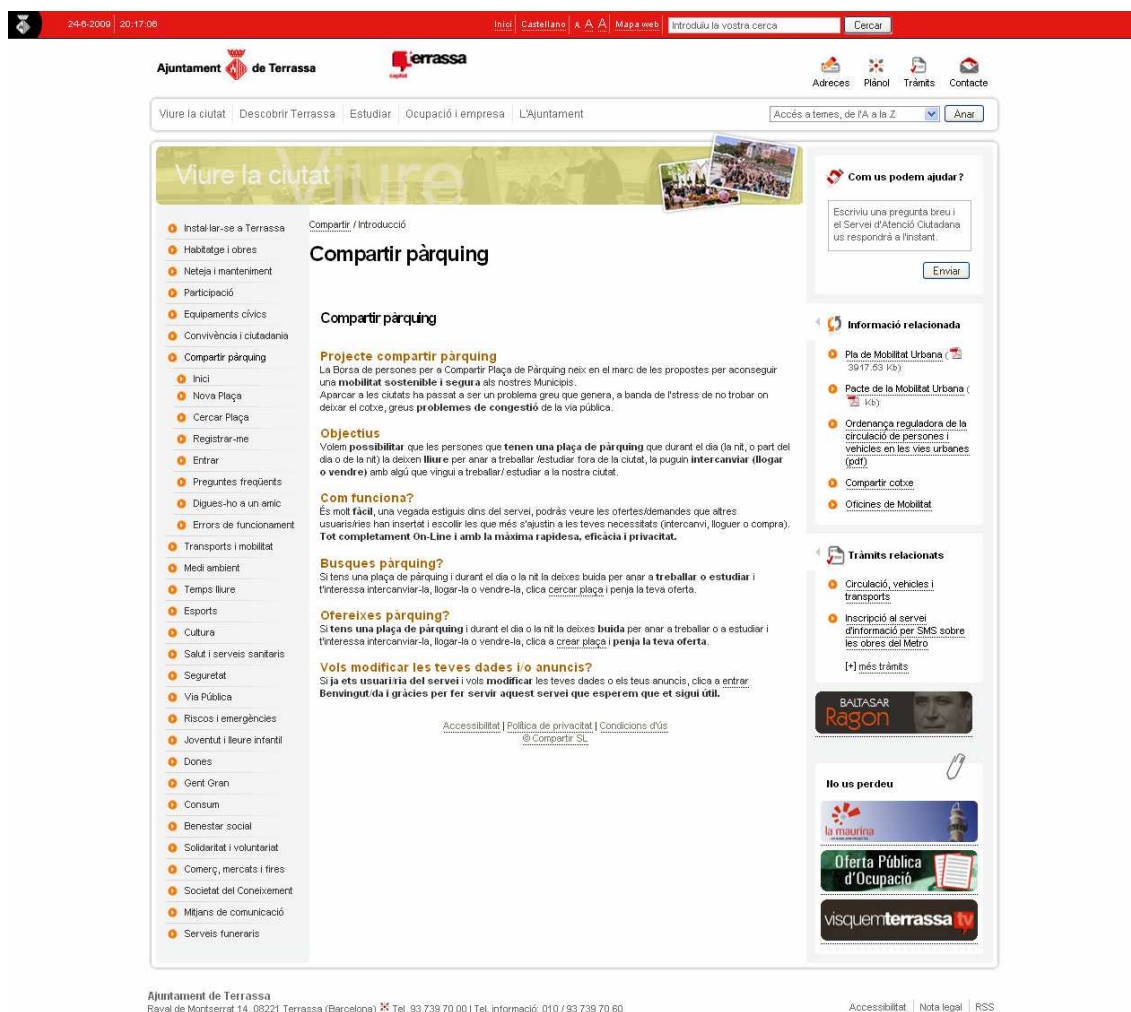
(Imagen 21 Pantalla con un mensaje de información estándar)

4.2 Integración en la Web del cliente

Por supuesto la cabecera de la aplicación no será añadida en la integración con la Web del cliente.

En caso que el cliente tenga unos colores o atributos que no concordaran con los colores hasta ahora utilizados se puede personalizar utilizando un fichero llamado “cobranded.css” que se colocará dentro de la carpeta del cliente.

Se ha hecho una integración con la Web del ayuntamiento de Terrassa y a continuación se muestra el resultado.



(Imagen 22 Pantalla de inicio del cliente Ayuntamiento de Terrassa)

Ajuntament de Terrassa

24-6-2009 20:42:46 Inici Castellano Mapa web Introduïu la vostra cerca Cercar

Adreces Plànol Tràmits Contacte

Viure la ciutat Descobrir Terrassa Estudiar Ocupació i empresa L'Ajuntament Accés a temes, de l'A a la Z Anar

Viure la ciutat

- Instal·lar-se a Terrassa
- Habitatge i obres
- Neteja i manteniment
- Participació
- Equipaments cívics
- Convivència i ciutadania
- Compartir pàrquing
- Compartir / Plaça / Nova Plaça
- Inici
- Nova Plaça
- Cercar Plaça
- Les Meves Places
- Els Meus Missatges
- Les meves dades
- Donar-se de baixa
- Sortir
- Preguntes freqüents
- Digueu-ho a un amic
- Error de funcionament
- Transports i mobilitat
- Medi ambient
- Temps lliure
- Esports
- Cultura
- Salut i serveis sanitaris
- Seguretat
- Via Pública
- Riscos i emergències
- Juventut i lleure infantil
- Dones
- Gent Gran
- Consum
- Benestar social
- Solidaritat i voluntariat
- Comerç, mercats i fires
- Societat del Coneixement
- Mitjans de comunicació
- Serveis funeraris

Compartir pàrquing

Nova Plaça

Alerta

Has de seleccionar una franja horària

Has de seleccionar un tamany de cotxe

Has de seleccionar el què vols fer amb la plaça

Ciutat: Actualitzar mapa

Adreça:

Codi Postal:

Fotografia de la plaça: Examinar...

Horari de disponibilitat de la plaça

Entrada necessària: ☐ Mat ☐ Tarda ☐ Nits ☐ Caps de setmana

Horari variable: ☐

Tamany del cotxe

Entrada necessària: ☐ Gran ☐ Mitjà ☐ Petit

¿Què vols fer amb la plaça?

Entrada necessària: ☐ Llogar ☐ Vendre ☐ Intercanviar

Preu de venda:

Preu de llogar (€/mes):

Comentari:

Accessibilitat | Política de privacitat | Condicions d'ús

@ Compartir SL

Com us podem ajudar?

Escriu una pregunta breu i el Servei d'Atenció Ciutadana us respondrà a l'instant.

Informació relacionada

- Pla de Mobilitat Urbana (3917.53 Kb)
- Pacte de la Mobilitat Urbana (1 Kb)
- Ordenança reguladora de la circulació de persones i vehicles en les vies urbanes (pdf)
- Compartir cotxe
- Oficines de Mobilitat

Tràmits relacionats

- Circulació, vehicles i transports
- Inscripció al servei d'informació per SMS sobre les obres del Metro
- (+) més tràmits

BALTASAR Ragon

No us perdeu

la maurina

Oferta Pública d'Ocupació

visquemterrasa

Ajuntament de Terrassa
Raval de Montserrat 14, 08221 Terrassa (Barcelona) ☎ Tel. 93 739 70 00 | Tel. informació: 010 / 93 739 70 60

Accessibilitat | Nota legal | RSS

(Imagen 23 Pantalla de nueva plaza del cliente Ayuntamiento de Terrassa)

The screenshot shows the website of the Ayuntamiento de Terrassa. The main navigation bar includes links like 'Inici', 'Castellano', 'A A A', 'Mapa web', and a search bar. The header features the 'Ajuntament de Terrassa' logo and 'terrassa' branding. Below the header, there's a secondary navigation bar with links like 'Viure la ciutat', 'Descobrir Terrassa', 'Estudiar', 'Ocupació i empresa', and 'L'Ajuntament'. The main content area is titled 'Compartir pàrquing' and shows a map of the city with numbered locations (1-5). Below the map, there's a list of parking spaces with details like address, availability, and user information. The right sidebar contains sections like 'Com us podem ajudar?', 'Informació relacionada', and 'Tràmits relacionats'. The footer includes contact information and links to 'Accessibilitat', 'Política de privacitat', and 'Condicions d'ús'.

(Imagen 24 Pantalla de listado de plazas de una
Búsqueda del cliente Ayuntamiento de Terrassa)

Cabe destacar que todas las pantallas explicadas anteriormente se mantienen exactamente igual, siempre y cuando no se haya personalizado mediante el fichero “cobranded.css”.

Capítulo 5

Pruebas

5.1 Buscar plaza

A continuación se muestra un informe con las pruebas de la sección de búsquedas de plazas.

5.1.1 Buscar ciudad/calle introduciendo valores aleatorios

No hay condiciones específicas para realizar esta prueba

Si se buscan letras aleatorias, o nombres que no existen se prevé que el sistema muestre una pantalla como la siguiente

Buscar

Ciudad

Campo necesario

jksdhfskjdf

Sin resultados

Campo necesario

Buscar

✓ Prueba superada correctamente.

5.1.2 Buscar ciudad/calle introduciendo caracteres maliciosos

No hay condiciones específicas para realizar esta prueba

Si se escriben caracteres maliciosos sobre el campo de ciudad o calle estos no lo procesan y al enviar el formulario con estos caracteres varias funciones del servicio Web se encargan de limpiarla de ataques.

Se prevé que tanto en caso de introducir código malicioso como números el sistema muestre un mensaje como el siguiente

Buscar

Ciudad

Campo necesario

'456'46"dflgñ<¿?

Sin resultados

Campo necesario

Buscar

✓ Prueba superada correctamente.

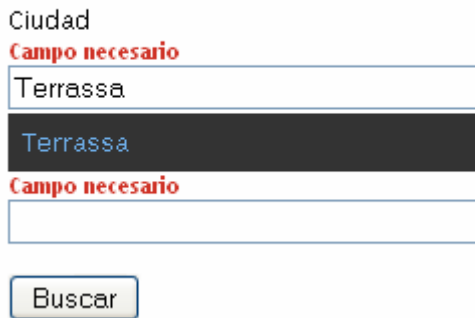
5.1.3 Buscar ciudad/calle con valores que realmente existen

No hay requisitos específicos para realizar esta prueba

Si se escribe correctamente la ciudad y/o la calle se muestra la sugerencia correspondiente.

Podemos prever que la aplicación mostrará un menú desplegable de sugerencias como el de la imagen

Buscar



Formulario de búsqueda con el título "Buscar". Hay dos campos de entrada, uno para "Ciudad" y otro para "Calle". El campo "Ciudad" contiene el texto "Terrassa" y muestra una lista desplegable con "Terrassa" como opción. Ambos campos tienen una etiqueta roja "Campo necesario" encima. Debajo de los campos hay un botón "Buscar".

NOTA: Se mostrará exactamente el mismo estilo de sugerencia para el campo de calle.

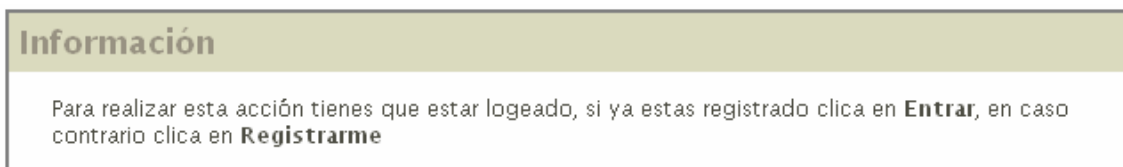
✓ Prueba superada correctamente.

5.1.4 Ver detalles de la plaza, ficha del usuario, enviar correo electrónico al creador de la plaza e intercambiar

Requisitos: no estar logeado

Una vez buscada la plaza si se intenta ver los detalles, ver la ficha de usuario del creador de la plaza, enviar un mensaje al creador de la plaza o se intenta intercambiar un mensaje nos informa que necesitamos estar logeados para poder hacer todas estas acciones. En caso de no estar registrado lo necesitaremos para poder logearnos.

Podemos prever que la aplicación mostrará una pantalla de aviso como la siguiente.



Pantalla de información con el título "Información". El texto dice: "Para realizar esta acción tienes que estar logeado, si ya estas registrado clica en **Entrar**, en caso contrario clica en **Registrarme**".

Si por lo contrario ya se está logeado se muestran las correspondientes pantallas de cada acción.

✓ Prueba superada correctamente.

5.2 Resultados de la búsqueda

Acto seguido se especifican los resultados de las pruebas de la sección de resultados de la búsqueda.

5.2.1 Detalle de la plaza modificando la URL manualmente, cambiando el identificador de la plaza

Requisitos: estar logeado

Si el usuario decide cambiar la dirección URL manualmente y probar números identificadores aleatorios de las plazas pueden pasar 2 cosas.

La primera es que introduzca números de plazas que existen, se prevé que se muestre la información de dichas plazas.

Lo segundo es que introduzca algún número de plaza que no exista, en este caso la previsión es un mensaje como este

Información
No hay datos sobre esta plaza

✓ Prueba superada correctamente.

5.2.2 Ficha de usuario modificando la URL manualmente, cambiando el identificador de usuario

Requisitos: estar logeado

Si el usuario decide cambiar la dirección URL manualmente y probar números identificadores de usuario pueden suceder 2 cosas.

La primera es que introduzca identificadores de usuario que existen, se prevé que se muestre la información pública del usuario.

Lo segundo es que introduzca algún identificador que no exista, en este caso la previsión es un mensaje como el siguiente

Información
No existe el usuario

✓ Prueba superada correctamente.

5.2.3 Enviar correo a usuario modificando la URL manualmente, cambiando el identificador de usuario

Requisitos: estar logeado

Si el usuario decide cambiar la dirección URL manualmente y probar números pueden suceder 2 cosas.

La primera es que introduzca identificadores de usuario que existen, se prevé que se redirija a la pantalla de enviar correo electrónico.

Lo segundo es que introduzca algún identificador que no exista, en este caso la previsión es un el mensaje como este

Información
No existe el usuario

✓ Prueba superada correctamente.

5.2.4 Intercambiar plaza modificando la URL manualmente, cambiando el identificador de la plaza

Requisitos: estar logeado

Si el usuario decide cambiar la dirección URL manualmente y probar números pueden suceder 2 cosas.

La primera es que introduzca identificadores de usuario que existen, se prevé que se muestre un listado con las plazas para intercambiar que tenga el usuario.

Lista de plazas para intercambiar

- ☐ Carrer de Pi i Margall, 200, Terrassa (8224)
- ☐ Carrer del Sindicat, Terrassa (8223)
- ☐ Camino de Can Ponsich, Sant Quirze del Vallès (8192)
- ☐ Carrer del Sindicat, 84, Terrassa (8223)

Continuar

Lo segundo es que introduzca algún identificador que no exista, en este caso la previsión es un mensaje como el siguiente

Información
No hay datos sobre esta plaza

✓ Prueba superada correctamente.

5.2.5 Intercambiar plaza sin tener ninguna plaza con la que intercambiar

Requisitos: estar logeado

Si se clica sobre intercambiar y el usuario no tiene creada ninguna plaza suya para intercambiar se prevé que se muestre el siguiente mensaje

Información
No tienes ninguna plaza para intercambiar

✓ Prueba superada correctamente.

5.2.6 Intercambiar plaza sin modificar la URL y teniendo una plaza para elegir

Requisitos: estar logeado

La previsión es que se muestre un listado con las plazas del usuario disponibles para intercambiar.

Lista de plazas para intercambiar

- ☐ Carrer de Pi i Margall, 200, Terrassa (8224)
- ☐ Carrer del Sindicat, Terrassa (8223)
- ☐ Camino de Can Ponsich, Sant Quirze del Vallès (8192)
- ☐ Carrer del Sindicat, 84, Terrassa (8223)

Continuar

Una vez seleccionada la plaza se espera un mensaje como el siguiente conforme se ha enviado un correo electrónico al usuario de la otra plaza con un enlace para intercambiar su plaza por la nuestra.

Información
Correo enviado correctamente

✓ Prueba superada correctamente.

5.3 Crear y/o modificar plaza

En este apartado se explican las pruebas realizadas sobre la sección de nueva plaza y modificar plaza.

5.3.1 Crear/modificar plaza sin estar logeado

Requisitos: no estar logeado

Se prevé que el sistema muestre un mensaje informando que para poder crear una plaza en caso de estar registrado se tiene que logear, y si aún no lo está debe registrarse primero.

Información

Para realizar esta acción tienes que estar logeado, si ya estas registrado clic en **Entrar**, en caso contrario clic en **Registrarme**



Prueba superada correctamente.

5.3.2 Crear/modificar plaza estando logeado y dejando valores sin introducir

Requisitos: estar logeado

Automáticamente después de enviar el formulario podemos prever que se muestre el siguiente mensaje especificando los campos que faltan por rellenar, si se clic sobre los campos que dice el mensaje que son erróneos nos lleva automáticamente sobre el elemento html que tenemos que cambiar.

Alerta

Falta introducir la ciudad
Falta introducir la dirección de la plaza
Tienes que elegir una franja horaria
Tienes que elegir un tamaño de coche
Tienes que elegir qué quieres hacer con la plaza

El mensaje de alerta dependerá del número de campos no correctos.



Prueba superada correctamente.

5.3.3 Crear/modificar plaza estando logeado y subiendo una imagen de más de 2MB

Requisitos: estar logeado

Como el tamaño máximo de subida de una imagen es de 2MB si se prueba subir un fichero de superior tamaño se prevé que se muestre este aviso

Alerta

El tamaño de la imagen no puede ser mayor que 2MB



Prueba superada correctamente.

5.3.4 Crear/modificar plaza estando logeado y subiendo un archivo que no sea extensión *.gif, *.jpg o *.bmp

Requisitos: estar logeado

Solamente se aceptan subir archivos que tengan las extensiones *.jpg, *.bmp y *.gif si se intenta subir cualquier otra extensión se hace la previsión que se mostrará el siguiente mensaje

Alerta

Sólo se aceptan ficheros bmp, jpg o gif



Prueba superada correctamente.

5.3.5 Crear/modificar plaza estando logeado y buscando dirección clicando en una zona de mar sobre el mapa de google

Requisitos: estar logeado

Dado que sobre el mar es imposible que haya una plaza de aparcamiento se prevé que el sistema haga caso omiso, y no deje poner una plaza en medio del mar.

Al no procesar la solicitud cuando se envía el formulario se prevé el retorno del siguiente mensaje

Alerta

Falta introducir la ciudad

Falta introducir la dirección de la plaza



Prueba superada correctamente.

5.3.6 Crear/modificar plaza estando logeado y buscando dirección clicando en una zona de donde no existan calles sobre el mapa de google

Requisitos: estar logeado

Puesto que toda plaza de aparcamiento tiene que tener una calle y un número si se selecciona alguna zona donde no hay calles, por ejemplo una montaña, se prevé que el sistema nos deje poner el marcador en el mapa pero cuando se envíe el formulario se espera un mensaje de aviso tal que así

Alerta

Falta introducir la dirección de la plaza



Prueba superada correctamente.

5.3.7 Código postal no numérico

Requisitos: estar logeado

El código postal tiene que ser un número entero si se introduce cualquier carácter que no sea número se prevé que la aplicación haga la búsqueda de la calle sobre google maps y recoja el código postal que retorna, haciendo así que este campo sea correcto.



Prueba superada correctamente.

5.3.8 Precio de venta y/o alquiler letras

Requisitos: estar logeado

El precio de venta y/o alquiler tiene que ser un valor numérico, puede ser un número no entero, con decimales pero tiene que ser un número.

En caso que se introduzcan caracteres no numéricos se espera que se muestre el siguiente error

Alerta

El precio de alquiler no es correcto

El precio de venta no es correcto



Prueba superada correctamente.

5.3.9 Precio de venta y/o alquiler número decimales con más de un punto o coma

Requisitos: estar logeado

En caso que se introduzcan valores numéricos con decimales pero por algún motivo tengan más de un punto o una coma se hace la previsión que el sistema muestre lo siguiente

Alerta

El precio de alquiler no es correcto
El precio de venta no es correcto

✓ Prueba superada correctamente.

5.3.10 Precio de venta y/o alquiler con número enteros

Requisitos: estar logeado

Un valor entero es uno de los valores correcto para estos campos, se espera que el sistema procese correctamente los datos sin mostrar ningún mensaje de error.

✓ Prueba superada correctamente.

5.3.11 Precio de venta y/o alquiler con número decimal con 1 sola coma o punto

Requisitos: estar logeado

Un valor real o float también es uno de los valores aceptados para este campo, se prevé que el sistema lo procesará sin mostrar ningún aviso.

✓ Prueba superada correctamente.

5.3.12 Crear plaza rellenando todos los campos necesarios

Requisitos: estar logeado

Si se han rellenado todos los campos necesarios con los valores correctamente se hace la previsión que el sistema realizará todas las comprobaciones y al estar todo correcto realizará la acción pertinente, una vez realizada dicha acción se mostraran estas pantallas

Información

La plaza se ha guardado correctamente

Información

La plaza se ha modificado correctamente

✓ Prueba superada correctamente.

5.4 Registrarme y mis datos

Aquí se listarán y expondrán las pruebas realizadas sobre las secciones de registrarme y mis datos.

5.4.1 Enviar el formulario sin rellenar los campos obligatorios

No hay requisitos específicos para realizar esta prueba

El hecho de enviar un formulario dejando campos obligatorios por rellenar hace esperar un mensaje de aviso como el siguiente

Alerta

Falta introducir el nombre
Falta introducir el apellido
El campo de tu correo electrónico no es correcto
El año de nacimiento no es correcto
Tienes que especificar un sexo
Falta introducir la ciudad

✓ Prueba superada correctamente.

5.4.2 Enviar el formulario poniendo un correo electrónico que no tenga el formato de correo (@ y .)

No hay requisitos específicos para realizar esta prueba

Si el campo de correo electrónico no tiene el formato correcto, una arroba y mínimo un punto, se prevé que la aplicación muestre un mensaje como el siguiente

Alerta

El campo de tu correo electrónico no es correcto

✓ Prueba superada correctamente.

5.4.3 Correo repetido

No hay requisitos específicos para realizar esta prueba

El correo electrónico es un elemento único, si se intenta registrar o modificar los datos de un usuario poniendo un correo que ya esté dado de alta en la aplicación se espera un mensaje como este

Alerta

Este correo electrónico ya está dado de alta

✓ Prueba superada correctamente.

5.4.3 Enviar formulario con contraseñas menores de 6 caracteres

No hay requisitos específicos para realizar esta prueba

Por motivos de seguridad la longitud mínima de la contraseña es de 6 caracteres, cualquier cadena mejor de ese tamaño nos hace prever un mensaje de aviso como el siguiente

Alerta

La contraseña deber ser de como mínimo 6 caracteres

✓ Prueba superada correctamente.

5.4.4 Enviar formulario con el campo contraseña y repetir contraseña distinta

No hay requisitos específicos para realizar esta prueba

Al enviar el formulario se comprueban las contraseñas si se han introducido distintas se puede esperar un aviso como este

Alerta

Las contraseñas no son iguales

✓ Prueba superada correctamente

5.4.5 Enviar formulario con el año de nacimiento inferior a 1950

No hay requisitos específicos para realizar esta prueba

La año de nacimiento mínimo es 1950, si se introduce un valor inferior a este se prevé un mensaje como el siguiente

Alerta

El año de nacimiento no es correcto

✓ Prueba superada correctamente.

5.4.6 Enviar formulario con el año de nacimiento igual al año actual

No hay requisitos específicos para realizar esta prueba

La edad mínima para utilizar el servicio Web es de 18 años, cualquier año de nacimiento que haga que la edad del usuario sea inferior a esta edad hace esperar un mensaje como el de a continuación

Alerta

El año de nacimiento no es correcto

✓ Prueba superada correctamente.

5.4.7 Enviar formulario siendo el año de nacimiento letras

No hay requisitos específicos para realizar esta prueba

En el supuesto caso que en el campo de año de nacimiento se introdujeran caracteres no numéricos nos haría esperar un aviso como el siguiente

Alerta

El año de nacimiento no es correcto



Prueba superada correctamente.

5.4.8 Enviar formulario siendo letras el código postal

No hay requisitos específicos para realizar esta prueba

En el caso que en el campo de código postal se introdujeran caracteres no numéricos nos haría prever un mensaje como el siguiente

Alerta

El código postal no es correcto



Prueba superada correctamente.

5.4.9 Enviar formulario siendo letras los campos de teléfono

No hay requisitos específicos para realizar esta prueba

Igual que en los casos anteriores el teléfono tiene que ser numérico, cualquier carácter no numérico nos haría esperar un mensaje informativo

Alerta

El teléfono fijo no es correcto

El teléfono móvil no es correcto



Prueba superada correctamente.

5.4.10 Enviar formulario con el campo de teléfono (móvil y/o fijo) con menos de 9 caracteres

No hay requisitos específicos para realizar esta prueba

Los campos de teléfono tienen que tener un mínimo de 9 dígitos, en caso contrario nos hace prever un mensaje de alerta como este

Alerta

El teléfono fijo no es correcto

El teléfono móvil no es correcto



Prueba superada correctamente.

5.4.11 Ir a la acción de registrarme estando logeado

Requisitos: estar logeado

Si se intenta forzar a la aplicación a ir a la sección de usuario y a la acción de registrarme estando ya logeados nos hace esperar una redirección a la pantalla que se estuviera viendo anteriormente.

✓ Prueba superada correctamente.

5.5 Baja de usuario

Todas las pruebas de la sección de baja de usuario.

5.5.1 Clicar sobre el botón de Sí de baja de usuario habiendo modificado manualmente la URL

Requisitos: estar logeado

En el caso que el usuario modifique la URL manualmente e introduzca el identificador de otro usuario se espera que el sistema le enseñare esta pantalla y no realizare la acción.

Información

No se ha podido borrar el usuario por favor vuelve a probarlo más tarde

✓ Prueba superada correctamente.

5.5.2 Clicar sobre el botón de Sí de baja de usuario sin modificar la URL manualmente

Requisitos: estar logeado

Si el usuario decide darse de baja pero en este último paso modifica de manera manual la URL para dar de baja a otro usuario se prevé que la aplicación no realizare tal acción y muestre el siguiente mensaje

Información

El usuario se ha borrado correctamente

✓ Prueba superada correctamente.

5.6 Entrar

Abanico de pruebas que cubren la sección de entrar

5.6.1 Introducir correo electrónico no válido

Requisitos: no estar logeado

Si el usuario se equivoca al introducir el correo electrónico se espera que el sistema informare con el siguiente mensaje, por medidas de seguridad el mensaje no especifica si es incorrecto el correo o la contraseña

Alerta

Nombre de usuario o contraseña incorrecta



Prueba superada correctamente.

5.6.2 Introducir contraseña no válida

Requisitos: no estar logeado

Si el usuario introduce una contraseña errónea se prevé que el sistema informare con el siguiente mensaje, por medidas de seguridad el mensaje no especifica si es incorrecto el correo o la contraseña

Alerta

Nombre de usuario o contraseña incorrecta



Prueba superada correctamente.

5.6.3 Introducir el correo electrónico todo en mayúsculas, todo en minúsculas o aleatoriamente mayúsculas y minúsculas

Requisitos: no estar logeado

Pese a que el registro se realice con el correo en minúsculas dado que es un campo único se espera que el sistema reconozca que el correo es el mismo y de acceso al sitio Web.



Prueba superada correctamente.

5.6.4 Introducir la contraseña toda en mayúsculas, toda en minúsculas o aleatoriamente mayúsculas y minúsculas

Requisitos: no estar logeado

Por razones de seguridad la contraseña distingue entre mayúsculas y minúsculas, si no se introduce la contraseña exacta que se puso cuando se registró se espera que le aplicación muestre un mensaje como el siguiente

Alerta

Nombre de usuario o contraseña incorrecta



Prueba superada correctamente.

5.6.5 Introducir los datos correctamente

Requisitos: no estar logeado

Si todos los datos son introducidos correctamente se espera que la aplicación realizare la petición y automáticamente redirija hacia el listado de plazas del usuario.



Prueba superada correctamente.

5.7 Contraseña olvidada

En este apartado se muestran las pruebas de la sección de contraseña olvidada.

5.7.1 Petición de contraseña olvidada con correo no registrado

Requisitos: no estar logeado

En caso que el correo electrónico que introduzcamos no esté dado de alta en el servicio Web se espera que se muestre el siguiente mensaje

Alerta

El correo electrónico no existe



Prueba superada correctamente.

5.7.2 Correo electrónico sin el formato de correo (@ y .)

Requisitos: no estar logeado

Si el campo de correo electrónico no tiene el formato correcto, una arroba y mínimo un punto, es de esperar que la aplicación muestre un mensaje como el siguiente.

Alerta

El correo electrónico introducido no es válido

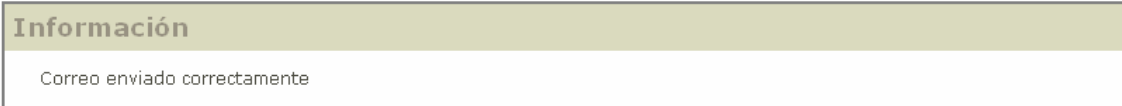


Prueba superada correctamente.

5.7.3 Petición de contraseña olvidada con correo real

Requisitos: no estar logeado

Si el servicio Web puede confirmar que el correo esta dado de alta, se prevé que se envíe un mensaje al correo que se acaba de introducir con un enlace para poder recuperar la contraseña, acto seguido se espera un mensaje tal como este



✓ Prueba superada correctamente.

5.8 Recuperar contraseña olvidada

5.8.1 Recuperar contraseña olvidada modificando algún dato del enlace enviado

Requisitos: no estar logeado

Si cualquier dato de la URL es cambiado manualmente se espera que el servicio Web muestre un error como el de la imagen



✓ Prueba superada correctamente.

5.8.2 Recuperar contraseña olvidada después de 24 horas de su petición

Requisitos: no estar logeado

Por medidas de seguridad el plazo límite para recuperar la contraseña es de 24 horas una vez pasado dicho periodo la petición de cambio queda anulada.

Si posteriormente a esas 24 horas intentamos realizar la petición se espera que la aplicación muestre el siguiente aviso



✓ Prueba superada correctamente.

5.8.3 Recuperar contraseña olvidada en un plazo menor a 24 horas

Requisitos: no estar logeado

Si no se ha cambiado manualmente ningún parámetro de la URL que llega al correo y el plazo de la petición de cambio de contraseña es menor a 24 horas se prevé que el servicio Web nos lleve a una pantalla en la que se ofrece un formulario para el cambio de contraseña

Recuperar contraseña

Contraseña

Repite la contraseña

Una vez introducida la contraseña pueden pasar varias cosas.

La primera que las contraseñas no sean iguales lo cual se espera que muestre esta pantalla.

Alerta

Las contraseñas no son iguales

La segunda que las longitudes sean muy cortas por lo que se prevé este otro mensaje

Alerta

La contraseña deber ser de como mínimo 6 caracteres

En el caso que las contraseñas cumplan estas restricciones se espera que se realice el cambio de contraseñas y se muestre este mensaje

Información

La contraseña se ha modificado correctamente



Prueba superada correctamente.

5.9 Mis plazas

Descripción y comportamiento de la batería de pruebas de la sección de mis plazas.

5.9.1 Descargar contrato en los todos los idiomas disponibles

Requisitos: estar logeado

Se espera que los documentos PDF se descarguen correctamente desde el navegador independientemente del idioma en el que se esté navegando.

✓ Prueba superada correctamente.

5.9.2 Cambiar estado

Requisitos: estar logeado

La petición de cambio de estado de una plaza se espera que se realice correctamente al pulsar sobre el botón de “Cambiar estado” y se prevé que el servicio Web muestre este mensaje



✓ Prueba superada correctamente.

5.9.3 Cambiar estado de una plaza modificando la dirección URL, seleccionando un número identificador de plaza aleatorio

Requisitos: estar logeado

Si el usuario decide cambiar la dirección URL manualmente y probar números pueden suceder 3 cosas.

La primera es que introduzca un número de plaza que exista y sea del usuario, se espera el mensaje de la siguiente imagen.



La segunda es que la plaza exista pero no sea del usuario se espera que se muestre el siguiente mensaje



La tercera es que introduzca algún número de plaza que no exista, en este caso se prevé que se muestre un mensaje como este

Información

No hay datos sobre esta plaza

5.9.3 Modificar datos de una plaza modificando la dirección URL poniendo un número inventado

Requisitos: estar logeado

Si el usuario decide cambiar la dirección URL manualmente y probar números pueden suceder 3 cosas.

La primera es que introduzca un número de plaza que exista y sea del usuario, se espera que se redirija a la pantalla de modificar datos de la plaza.

La segunda es que la plaza exista pero no sea del usuario se espera que se muestre el siguiente mensaje

Información

No puede modificar esta plaza porque no es tuya

La tercera es que introduzca algún número de plaza que no exista, en este caso se prevé que se muestre un mensaje como este

Información

No hay datos sobre esta plaza



Prueba superada correctamente.

5.9.4 Renovar unan plaza modificando la dirección URL poniendo un número inventado

Requisitos: estar logeado

Si el usuario decide cambiar la dirección URL manualmente y probar números pueden suceder 3 cosas.

La primera es que introduzca un número de plaza que exista y sea del usuario, se prevé que se realice la renovación de la plaza y se espera también que se muestre el mensaje de la imagen

Información

La plaza se ha renovado correctamente

La segunda es que la plaza exista pero no sea del usuario se prevé que se muestre el siguiente mensaje

Información
Esta plaza no es tuya

La tercera es que introduzca algún número de plaza que no exista, en este caso se espera un mensaje tal que así

Información
No hay datos sobre esta plaza



Prueba superada correctamente.

5.9.5 Borrar una plaza modificando la dirección URL poniendo un número inventado

Requisitos: estar logeado

Si el usuario decide cambiar la dirección URL manualmente y probar números pueden suceder 3 cosas.

La primera es que introduzca un número de plaza que exista y sea del usuario, se espera que la aplicación redirija a una pantalla donde se pregunte si se está seguro que se quiere borre la plaza

Borrar

Estas seguro que quieres dar de baja esta plaza?

Sí	No
----	----

Si en dicha pantalla se clicca que si se prevé que se muestre el mensaje

Información
La plaza se ha borrado correctamente

Si se modifica la URL del Si poniendo otro identificador que no sea nuestro se espera el mensaje

Información
Esta plaza no es tuya

La segunda es que la plaza exista pero no sea del usuario se prevé mostrar el siguiente mensaje

Información
Esta plaza no es tuya

La tercera es que introduzca algún número de plaza que no exista, en este caso la previsión es el siguiente mensaje

Información

No hay datos sobre esta plaza

✓ Prueba superada correctamente.

5.9.6 Pulsar sobre los enlaces de datos de la plaza, modificar, renovar y borrar plaza sin modificar nada de la URL

Requisitos: estar logeado

Se realizaran las acciones requeridas por cada enlace ya que se asegura que los parámetros del servicio Web son los correctos.

Cabe esperar que se realice la acción correspondiente sin ningún tipo de error ni aviso.

✓ Prueba superada correctamente.

5.10 Mis mensajes

Aquí se explicaran las pruebas realizadas sobre la sección de mis mensajes.

5.10.1 Borrar mensaje cambiando identificador del mensaje

Requisitos: estar logeado

Si el usuario decide cambiar la dirección URL manualmente y probar números pueden suceder 3 cosas.

La primera es que introduzca un identificador de mensaje que exista y sea del usuario, se prevé que se borre el mensaje automáticamente y se espera mostrar la pantalla

Información

El mensaje se ha borrado correctamente

La segunda es que el identificador de mensaje exista pero no sea del usuario se espera mostrar el siguiente mensaje

Información

No puede borrar este mensaje porque no es tuyo

La tercera es que introduzca algún identificador de mensaje que no exista, en este caso se espera mostrar un mensaje como este

Información

El mensaje no existe

✓ Prueba superada correctamente.

5.10.2 Pulsar sobre el enlace de ficha de usuario, enviar correo electrónico, o borrar

Requisitos: estar logeado

Es de esperar que las acciones se realicen correctamente.

✓ Prueba superada correctamente.

5.11 Administración, nuevo cliente y/o modificar cliente

Listado de pruebas realizadas para la validación de la sección de administración.

5.11.1 No todos los campos obligatorios rellenos

Requisitos: estar logeado y tener permisos de administración o trabajador

En caso que el usuario se deje alguno de los campos obligatorios por rellenar se espera que se muestre un listado de los campos que no son correctos para que los introduzca bien.

La pantalla de aviso se prevé como la siguiente, dependerá del número de campos que no sean correctos.

Alerta

- Falta introducir el nombre
- Falta introducir el nombre de la carpeta
- Falta introducir la dirección URL de compartir
- Falta introducir la dirección Web del cliente
- Falta introducir la ciudad
- Tienes que elegir el idioma por defecto que tendrá del cliente
- Falta introducir los idiomas disponibles del cliente
- Tienes que elegir la disponibilidad
- Tienes que elegir el tipo de entidad
- Tienes que elegir si quieres activar o no las estadísticas

✓ Prueba superada correctamente.

5.11.2 Campo urlCobranded sin extensión del dominio

Requisitos: estar logeado y tener permisos de administración o trabajador

El campo urlCobranded debe contener el nombre de la carpeta seguido de punto y posteriormente el dominio con la extensión.

Si por algún motivo el dominio y la extensión no se encuentran en este campo es de esperar que se muestre el siguiente aviso

Alerta

La dirección URL de compartir no es correcta



Prueba superada correctamente.

5.11.3 Ciudad inventada

Requisitos: estar logeado y tener permisos de administración o trabajador

Si la ciudad que introducimos no se encuentra en la tabla de ciudades de nuestra Base de datos se espera que se muestre el mensaje

Alerta

No se ha encontrado la ciudad

En caso contrario se prevé que la petición supere exitosamente esta prueba.

NOTA: la tabla ciudades contiene todas las ciudades y pueblos de España.



Prueba superada correctamente.

5.12 Díselo a un amigo

Pruebas de la sección díselo a un amigo.

5.12.1 No todos los campos obligatorios rellenos

No hay condiciones específicas para realizar esta prueba

Si alguno de los campos obligatorios no están rellenos se prevé que el sistema Web muestre un mensaje como el siguiente. El mensaje dependerá de las opciones que falten por rellenar o sea incorrectas.

Alerta

Falta introducir el nombre
El campo de tu correo electrónico no es correcto
Falta introducir el nombre de tu amigo
El campo del correo electrónico no es correcto



Prueba superada correctamente.

5.12.2 Correos electrónicos sin el formato de correo (@ y .)

No hay condiciones específicas para realizar esta prueba

Si alguno de los dos campos de correo electrónico no tiene el formato correcto, una arroba y mínimo un punto, se espera que la aplicación muestre un mensaje como el siguiente.

Alerta

El campo de tu correo electrónico no es correcto
El campo del correo electrónico no es correcto

✓ Prueba superada correctamente.

5.13 Errores de funcionamiento

Otra batería de pruebas esta vez para la sección de errores de funcionamiento.

5.13.1 No todos los campos obligatorios rellenos o seleccionados

No hay condiciones específicas para realizar esta prueba

Si alguno de los campos obligatorios no están rellenos se prevé que el sistema Web muestre un mensaje como el siguiente. El mensaje dependerá de las opciones que falten por rellenar o sea incorrectas.

Alerta

Falta introducir la descripción del problema
Falta introducir la descripción detallada
Falta introducir el nombre
El correo electrónico introducido no es válido

✓ Prueba superada correctamente.

5.13.2 Correos electrónicos sin el formato de correo (@ y .)

No hay condiciones específicas para realizar esta prueba

Si el campo de correo electrónico no tiene el formato correcto, una arroba y mínimo un punto, es de esperar que la aplicación muestre un mensaje como el siguiente.

Alerta

El correo electrónico introducido no es válido

✓ Prueba superada correctamente.

5.14 Comprobaciones Generales

Conjunto de pruebas generales extensibles a toda la aplicación, estas pruebas sirven en todas las secciones.

5.14.1 Código de algún lenguaje en los campos de texto

No hay condiciones específicas para realizar esta prueba

La inyección de código malicioso o XSS en cualquier campo es procesada al enviar el formulario, se espera que cuando la aplicación reciba los datos los limpie de todos estos caracteres no gratos y retorne los valores limpios.



Prueba superada correctamente.

5.14.2 Modificar sobre la URL el valor de la paginación poniendo un número mayo al máximo

No hay condiciones específicas para realizar esta prueba

Se espera que la aplicación calcule la última página y redirija allí sin hacer caso al número de página mayor introducido.



Prueba superada correctamente.

5.14.3 Modificar sobre la URL el valor de la paginación poniendo un número negativo

No hay condiciones específicas para realizar esta prueba

Se espera que la aplicación haga caso omiso a dicho hecho y muestre la primera página.



Prueba superada correctamente.

5.14.4 Modificar sobre la URL el valor de la paginación poniendo valores no numéricos

No hay condiciones específicas para realizar esta prueba

Se espera que la aplicación haga caso omiso a dicho hecho y muestre la primera página.



Prueba superada correctamente.

5.14.5 Cambiar idioma

No hay condiciones específicas para realizar esta prueba

Se espera que se realice correctamente el cambio de idioma en cualquier lugar de la aplicación sin ningún tipo de problema o aviso.

✓ Prueba superada correctamente.

5.14.6 Accesibilidad Web

Se espera que la aplicación supere el test de accesibilidad Web, mínimo la doble A.

Se han utilizado las siguientes páginas para comprobar la accesibilidad.

<http://www.tawdis.net> y <http://www.sidar.org/hera/index.php.es>

✓ Prueba superada correctamente.

5.14.7 Validación del libro de estilos

Se prevé que el servicio Web supere los test de validación de css del Consorcio World Wide Web (W3C).

Se ha utilizado la Web oficial de validación del W3C enlazada en el siguiente hipervínculo.

<http://jigsaw.w3.org/css-validator/validator?uri=http%3A%2F%2Fcompartir.cat%2F&profile=css21&usermedium=all&warning=1&lang=es>

✓ Prueba superada correctamente.

5.14.7 Validación de XHTML

Se espera que la aplicación supere el test de validación de XHTML 1.0 Strict.

Para la comprobación se ha utilizado la siguiente Web.

[http://validator.w3.org/check?uri=http%3A%2F%2Fcompartir.cat%2F&charset=\(detect+automatically\)&doctype=Inline&group=0&user-agent=W3C_Validator%2F1.654](http://validator.w3.org/check?uri=http%3A%2F%2Fcompartir.cat%2F&charset=(detect+automatically)&doctype=Inline&group=0&user-agent=W3C_Validator%2F1.654)

✓ Prueba superada correctamente.

Capítulo 6

Codificación

6.1 Distribución de los ficheros

Los ficheros están distribuidos como se muestra en el árbol que vemos a continuación.



6.2 Ficheros .htacces

Los ficheros .htaccess son llamados por el servidor apache al entrar en un directorio.

Actualmente la aplicación consta de 2 tipos de estos ficheros.

Uno colocado en la raíz que se encarga de redirigir todo el flujo de las peticiones hacia el índice.

El otro está colocado en la carpeta de “fotosplazas” este .htaccess se encarga de evitar los llamados hotlinks. El .htaccess evita que Webs externas puedan enlazar las fotografías de las plazas de los usuarios.

6.3 Ficheros de configuración

El servicio Web consta de 2 archivos de configuración.

Ambos ubicados en la carpeta “motor/nucleo”.

El primer archivo es “dbconfig.inc.php” es el que contiene toda la información referente a la conexión con la Base de Datos.

El segundo archivo es “Configuracion.inc.php” contiene la información de la configuración de la aplicación, datos necesarios durante la ejecución como idiomas en los que está disponible la plataforma, ruta de los css, javascript entre otros.

6.4 FrontController

El FrontController es el que se encarga de recibir todos los datos y las peticiones y distribuirlas a los controladores adecuados.

Es como una centralita telefónica.

Además de redirigir las peticiones aplica una serie de medidas de seguridad para asegurar la integridad de los datos.

6.5 Objetos

Existe una carpeta llamada “objetos” en la cual están los objetos que se necesitan para el correcto funcionamiento de la aplicación.

Los objetos son clases de php, cada objeto tiene definido únicamente sus variables, el tipo que son y si necesitan algún valor por defecto.

6.6 Alias

Son un conjunto de ficheros que contienen la traducción de todas las secciones y acciones de la URL.

Estos ficheros son llamados desde los controladores de cada sección.

6.7 Controladores

Los controladores son las clases que se encargan de realizar los cálculos y las acciones que requiere el usuario.

Cada sección tiene su controlador y todas ellas heredan de un controlador general con funciones y utilidades comunes para todos los controladores.

Estas funciones son necesarias para los controladores hijos.

Un ejemplo es el hecho de añadir un fichero css, el método que añade el fichero está declarado en la clase padre de los controladores de esta manera si en una sección o acción en concreto necesitamos cargar un css determinado se puede utilizar sin necesidad de repetir el código en todas las clases hijas.

6.8 Modelos

Los modelos son los encargados de interactuar con la base de datos.

El modelo utiliza la clase “BaseDeDatos” para acceder nutrirse de una serie de funciones necesarias para el acceso y resolución de las consultas sobre la base de datos mysql.

Cabe destacar que la clase “Modelo” y “BaseDeDatos” tienen independencia una de la otra para facilitar la exportación a otros proyectos y simplificar las ampliaciones.

Los modelos son opcionales, existe un modelo genérico que contiene funciones genéricas como guardar, modificar, buscar o borrar de la base de datos.

Hay casos en los que estas opciones no son suficiente por ese motivo se pueden crear tantos modelos como sean requeridos.

Los modelos tiene que heredar de la clase padre “Modelo” para poder utilizar todas las funcionalidades de esta así como acceder a la clase “BaseDeDatos”.

Un ejemplo de la necesidad de un modelo es el necesitar una consulta que necesite más de una tabla.

6.9 Vista

La vista es la encargada de procesar los ficheros *.html para rellenarlos con los datos necesarios.

Como norma general el controlador procesa la acción, en caso de necesitar datos de la Base de datos llama al modelo y posteriormente envía todos estos datos a la vista para que esta los formatee de manera correcta y bonita.

La vista tiene también métodos para cargar en un buffer los ficheros *.html de tal manera que no se tengan que hacer impresiones de pantalla parciales.

Se guarda toda la información en el buffer y se muestra todo como un bloque.

Este buffer también es necesario para poder incluir ficheros *.html des de cualquier acción de cualquier sección.

6.10 Utilidades

Se ha creado una serie de utilidades que tienen total independencia del resto de clases.

6.10.1 CobrandedModel

Es la clase que se encarga de personalizar las páginas Web de los clientes para el servicio Web para compartir plazas de aparcamiento tenga un aspecto idéntico a la Web original del cliente.

6.10.2 Email

Esta clase se encarga de preparar los correos electrónicos que se enviarán rellenando destinatarios y otros campos necesarios.

6.10.3 Google

Esta otra clase es la que interacciona con google maps para recibir información sobre las direcciones, códigos postales entre otros.

6.10.4 Gráficos

Clase que implementa la librería de google chart facilitando la creación y uso de gráficos verticales, horizontales, de comparación y circulares entre otros.

6.10.5 LOG

Clase que permite guardar información sobre el estado de las peticiones que realiza el servicio Web, su principal cometido es la de facilitar la detección de errores durante la fase de desarrollo.

6.10.6 PHPMailer

Por último esta clase se encarga de enviar físicamente los mensajes de correo electrónico que han sido preparados por la utilidad 6.10.2 Email.

6.11 Seguridad

El servicio Web consta de varios protocolos para evitar SQL inyection, ataques XSS entre otros.

Cuando se realiza una petición se centraliza la información en el index, este index crea el FrontController y este distribuye la acción a realizar al controlador adecuado.

Una vez la información está en el FrontContrller se realizan las tareas de limpieza de todos y cada uno de los datos enviados por el usuario.

La aplicación también está provista de métodos para evitar el hotlink.

Las fotografías de las plazas de los cliente no podrán ser enlazadas desde sitios Web externos al dominio que se utilice, si se enlaza una imagen alojada en nuestro servidor en otro dominio diferente al que utilizamos se mostrará el logotipo del servicio.

A más a más de lo anteriormente explicado la está dotada de métodos para permitir y/o restringir el acceso a determinadas IP's o grupos de IP's tanto a nivel general de la aplicación como a nivel de cada cliente específico.

6.12 Personalización para un cliente

La mayoría de los datos de un cliente se pueden personalizar, todos los controladores son personalizables así como los layouts.

En el caso de los controladores se tiene que crear un controlador llamado "controladorSECCIONCobranded" donde SECCION será el nombre de la sección que se quiere personalizar, este fichero debe extender de "controladorSECCION".

En el archivo "controladorSECCIONCobranded" se podrán definir las peculiaridades del cliente.

Para personalizar los libros de estilo, javascript y layouts basta con emular la dirección de carpeta que hay en la ruta.

EJ:

Añadir el layout "crear.html" que está en la ruta "layout/usuario"

Basta con crear dentro de la carpeta del cliente una carpeta llamada usuario y allí introducir la modificación del archivo "crear.html"

Además de estas personalizaciones cada cliente tiene que tener una serie de archivos llamados "indexNUMEROIDIOMA.php" (NUMEROIDIOMA es el código numérico que define un idioma) el cliente tendrá tantos ficheros de este tipo como idiomas en los que esté disponible el servicio.

EJ: Cliente que tiene dos idiomas, catalán y castellano.

Tendrá dos “index”, uno llamado “index1.php” (catalán) y otro llamado “index2.php” (castellano) en dichos ficheros se especifican parámetros de la personalización como la dirección de donde se importará la página Web del cliente o los trozos que se añadirán al servicio Web para simular la integración.

6.13 Crear un módulo nuevo

Crear nuevos módulos es muy sencillo.

- 1- En caso que se necesite una tabla en la Base de Datos se crea.
- 2- Si hay algún campo que deba tener un valor por defecto se crea un objeto con los campos necesarios y los valores prefijados.
- 3- Se crea un archivo en la carpeta “/motor/controladores” llamado “controladorNOMBRESECCION.inc.php” y que extienda de la clase “Controlador”. Es importante que el nombre de la sección con el que se crea el fichero sea el nombre original. En este controlador se declara y definen todas las acciones que podrá realizar esta sección.
- 4- Se edita el fichero “Alias.inc.php” de la carpeta “/motor/alias” añadiendo las traducciones de la nueva sección.
- 5- Se crea un fichero en la ruta “/motor/alias” llamado “controladorNOMBRESECCIONAlias.inc.php”. Es importante que el nombre de la sección sea el nombre original. Se rellena este fichero con el nombre y la traducción a los diferentes idiomas de todas las funciones y métodos que se han declarado en el “controladorNOMBRESECCION.inc.php”.
- 6- En la carpeta “layouts” se tiene que crea una carpeta con el nombre de la sección que estamos creando y dentro de esta se crean los fichero *.html que darán formato a los datos. Por defecto el sistema cuando ejecuta una acción busca el layout con el mismo nombre que la acción, si no se cambia este hecho se tiene que crear tantas plantillas *.html como acciones hayan en el controlador.
- 8- En caso que se necesiten ejecutar sentencias SQL entre varias tablas se tendrá que crear un archivo llamado “ModeloNOMBRESECCION.inc.php” en la ruta “/motor/modelos” que extienda de la clase “Modelo”. Dentro de este modelo se declaran todas las sentencias necesarias.

Siguiendo estos sencillos pasos se tendrá en muy poco tiempo un módulo nuevo creado.

6.14 Patrones utilizados

Durante el desarrollo de la aplicación se han ido utilizando diferentes patrones.

Todo el proyecto está basado en el patrón Modelo Vista Controlador (MVC).

Dicho modelo separa las capas de presentación de las de acceso a datos y de las de controles generales.

Este hecho facilita el desarrollo y mantenimiento.

Los pasos de este modelo son los siguientes:

1. El usuario interactúa con la interfaz de usuario de alguna forma (por ejemplo, el usuario pulsa un botón, enlace, etc.)
2. El controlador recibe la notificación de la acción solicitada por el usuario. El controlador gestiona el evento que llega, frecuentemente a través de un gestor de eventos (handler) o callback.
3. El controlador accede al modelo, actualizándolo, posiblemente modificándolo de forma adecuada a la acción solicitada por el usuario. Los controladores complejos están a menudo estructurados usando un patrón de comando que encapsula las acciones y simplifica su extensión.
4. El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz apropiada para el usuario donde se refleja los cambios en el modelo. El modelo no debe tener conocimiento directo sobre la vista. Un objeto vista puede registrarse con el modelo y esperar a los cambios, pero aun así el modelo en sí mismo sigue sin saber nada de la vista. En la implementación utilizada en el proyecto actual la vista no tiene acceso directo al modelo, dejando que el controlador envíe los datos del modelo a la vista.
5. La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente.

Otros patrones utilizados son el Singleton para asegurar que sólo hay una instancia de la Base de Datos, el composite para crear objetos personalizados a partir de funcionalidades básicas y el strategy por ejemplo para el tema de las estadísticas.

6.15 Problemas surgidos

En este capítulo se explican los problemas que han ido saliendo durante todas las fases del proyecto así como su solución en caso de haberla.

Los principales problemas surgieron en la fase de desarrollo.

6.15.1 Concentrar todo el flujo de la aplicación en un único punto

Para simplificar el desarrollo todas las peticiones pasan por el fichero “index.php” allí se crea el “FrontController” que es la clase encargada de gestionar los flujos de la aplicación.

Para conseguir que todas las peticiones lleguen al “FrontController” ha sido necesario utilizar un fichero “.htaccess” con scripts de apache.

El fichero se colocó en la raíz del servidor así todas las peticiones pasaran por aquí.

En el “.htaccess” sólo especificamos que cualquier petición vaya al “index.php”, a no ser que la petición sea un archivo con extensión *.html, *.gif, *.jpg, *.png, *.ico, *.js o *.css y exista físicamente en el servidor.

Si el archivo tiene cualquiera de las extensiones anteriores pero no existe en el servidor se redirigirá igualmente al “index.php”.

Esta premisa es necesaria ya que sino sería imposible por ejemplo visualizar una imagen, siempre nos redirigiría al “index.php”.

También es necesario porque cuando enviamos una petición al servidor y éste nos retorna la pagina con sus imágenes, css, js, etc se redirigiría al “index.php”, cargándolo tantas veces como elementos tenga la página.

6.15.2 Mostrar por pantalla únicamente cuando se quiera

Otro problema era el hecho de gestionar la petición como un bloque sin tener que ir mostrando los contenidos parciales.

Dado que mostrar información por pantalla es costoso a nivel de tiempo, se ha evitado ir mostrando información parcial de la petición.

Se ha guardado toda la información a visualizar en un buffer.

Una vez se acaba de procesar toda la petición se imprime este buffer. De esta manera sólo se imprime la información una sola vez y se reduce el tiempo total de procesamiento.

6.15.3 Propiedades de css interpretadas diferentes por los distintos navegadores

Como es bien conocido las propiedades css que deberían ser estándar para todos los navegadores no lo son.

Firefox, Opera y Chrome interpretan la mayoría de las propiedades css de igual manera mientras que Internet Explorer no.

Todos los Internet Explorer por debajo de la versión 7, no incluida, son los que más difieren.

Dado que la aplicación tiene que superar exitosamente las pruebas de validación del W3C referente a css no se han utilizado hacks.

Para solucionar este problema se ha creado un archivo css con las propiedades que difieren y se carga única y exclusivamente para los navegadores que lo necesitan.

De esta manera si visualizamos con IE6 obtendremos una correcta visualización sin afectar al resto de navegadores que dichas propiedades no las interpretarían.

6.15.4 Automatización de los elementos de html

Se han creado unas funciones específicas para generar automáticamente elementos de html que contienen varias opciones como por ejemplo los Selects, Radio Buttons o los Check Buttons.

A esta funciones se les especifica #ID y automáticamente hace una consulta a la BBDD y retornado los elementos con el estilo del elemento html seleccionado, se pueden marcar valores por defecto o restringir la aparición de las opciones.

Al automatizar surgía el problema que los valores que retornaba siempre eran numéricos, necesarios para insertarlos en la Base de Datos pero no correctos para mostrarlos en las fichas.

Para solucionarlo se ha creado una función para codificar o decodificar estos valores numéricos.

6.15.5 Cambiar la URL de idioma

En todo momento se puede cambiar de idioma, para hacer posible esto se incluyó en el "index.php" una variable de Sesión que guardara la dirección URL, al pulsar sobre le idioma que se desea cambiar se carga el nuevo idioma y se recarga la página que hemos guardado anteriormente.

El problema era que al cambiar de idioma no traducía la URL al nuevo idioma. Para hacer posible esto se crearon unos Alias, los alias son simplemente nombres ficticios que hacen referencia a un nombre de una clase o función real.

La URL está estructurada así: dominio/sección/acción/parametros

El “index.php” recibe la petición y trata la URL. Siempre que se recibe una petición se traduce la sección y la acción al idioma original, es decir, se obtienen los nombres originales de las clases y funciones.

Internamente trabaja con los nombre originales y sólo se traducen al hacer la redirección.

La función que hace las traducciones se utiliza en el caso de redirecciones y/o para traducir todos los enlaces (hipervínculos) que haya en nuestra Web.

Así siempre que se pulse sobre un enlace el usuario podrá ver la dirección URL en el idioma que tiene seleccionado.

Lo primero que hace esta función es traducir la sección.

Carga el archivo “Alias.inc.php” y allí se busca la traducción de la sección.

Se comprueba cual es el nombre original de la sección para poder cargar su controlador.

Acto seguido se carga el controlador de la sección y se llama a una función que carga los alias de dicha sección.

Una vez hemos cargado los alias de la sección ya se puede hacer la traducción.

Una vez traducida la sección y la acción se le añaden los parámetros en caso que tengan y se retorna todo como una cadena de texto.

6.15.6 Alias

En el apartado anterior (*8.1.5 Cambiar la URL de idioma*) se ha hablado de los alias.

En un principio tanto los alias de las secciones como de las acciones se guardaron en la Base de Datos con un “_” delante para poder distinguirlos. Con ese patrón sería posible cambiar los idiomas de forma rápida y eficaz.

Después de varias pruebas se optó por cambiar el método.

El motivo del cambio es que al no saber que idioma se iba a utilizar se tenían que cargar todos, al cargarlos todos se tenía que crear una función que fuera recorriéndolos uno a uno, separándolos según el idioma y guardándolos en un array o una estructura.

El proceso de recorrer todas las traducciones era muy lento y costoso.

Se optó por generar un fichero para cada sección con las traducciones de las acciones de esa sección y uno global para el nombre de cada sección.

El fichero global tiene la traducción de todas las secciones (clases) mientras que el específico de cada sección tiene la traducción de todas las acciones (funciones de la clase).

Ambos tipos de ficheros tiene la misma estructura, hay un array donde se define como clave el nombre original y como valor el nombre que el usuario verá.

6.15.7 Sección sin Acción

El sistema está pensado para que siempre que haya una sección haya también una acción. Una sección no es nada, es un simple contenedor, se necesita especificar una acción para poder realizar cualquier petición.

Pese a que el sistema web siempre pone la sección y la acción es posible que por algún motivo el usuario decida eliminar la acción, si esto pasaba, automáticamente la aplicación no sabía que tenía que hacer y se quedaba parada.

Con la finalidad de solucionar este contratiempo se ha añadido a cada sección una variable con el nombre de la acción por defecto.

En caso que no se especifique la acción automáticamente el servicio web cargará la acción que tiene por defecto la sección en la que se encuentra.

6.15.8 Contador de visitas

Cada vez que se visita una página se actualiza un contador donde se especifica el número de visitas y de páginas vistas.

Utilidad necesaria para las estadísticas de los clientes.

El problema surge cuando un robot de un buscador entra en la Web para indexarla.

El robot hace muchísimas peticiones a la página con la finalidad de poder indexarla en los buscadores, este hecho hacía que los datos fueran desorbitados y confusos.

Dicho problema se solucionó haciendo un filtro.

El filtro no puede bloquear los robos ya que sino los buscadores no sabrían que la página existe.

Cuando alguien se conecta al servicio Web se comprueba que el campo de "HTTP_USER_AGENT" (dicho campo nos da información sobre el navegador, sistema operativo, etc) no tenga palabras que denoten que es un robot.

Las palabras son bot, spider, crawl, search entre otras.

Si contiene esta serie de palabras no se cuenta las visitas ni las paginas vistas, en cualquier otro caso sí.

6.15.9 Array a Objeto (case sensitive)

Otro de los problemas que surgió es que al convertir un array en un objeto los nombres tenían que ser idénticos.

Cuando se crea un objeto lleva intrínseco a su creación una serie de métodos mágicos (los métodos mágicos son unos métodos que son llamados en situaciones especiales).

Uno de los métodos mágicos es que en caso de intentar asignar una variable de un objeto y que ésta no exista en el objeto se crea (método mágico __set).

Cuando se intentaba pasar un array a Objeto y diferían los nombres por cuestión de mayúsculas se creaba una variable nueva dentro del Objeto, este hecho hacía imposible tratar correctamente los objetos.

Ejemplo:

```
array = array( 'Nombre' => 'dani', 'Universidad' => 'EUI' );
```

```
Objeto = [  
    public $nombre = “;  
    public $Universidad = “;  
];
```

Al intentar convertir el array a este tipo de Objeto quedaba el resultado:

```
Objeto = [  
    public $nombre = “;  
    public $Universidad = 'EUI';  
    public $Nombre = 'dani';  
];
```

Como vemos hay dos variables que son iguales salvo por la mayúsculas.

Para solucionar esto se utilizó una función que comprobara que dentro del objeto hubiera una variable con el mismo nombre, sin importar las mayúsculas y a su vez retornara el nombre para poder hacer la asignación del valor del array.

6.15.10 Automatización de guardado y modificación en la Base de Datos

Cuando se pide a la aplicación que guarde o modifique una información en la Base de Datos se hace utilizando el nombre del objeto como tabla (en caso que no se especifique lo contrario) y las variables como los campos de las tablas.

En ciertos momentos era necesario crear variables que en la Base de datos no estaban contempladas, por ejemplo cuando se recoge la información de una alta de un usuario está el campo de repetir contraseña que únicamente sirve de control.

Dado que el objeto tenía más variables que campos, que habían variables que no coincidían y que la sentencia se hacía sin ninguna restricción no funcionaba.

Para paliar este problema en los métodos de guardar y modificar antes de generar la sentencia automática se hace una petición a la base de datos pidiéndole un listado de los campos de esa tabla.

Acto seguido se asignan a estos campos el contenido de la variable del objeto que tiene su mismo nombre.

De esta manera se tiene controlado en todo momento que las sentencias serán correctas y damos a los objetos independencia de la Base de Datos.

6.15.11 Comprobar comillas

El hecho de poder facilitar el trato de las sentencias sql nos lleva al problema del control de las comillas doble y las comillas simples.

Por defecto, cuando se realiza la sentencia automática se ponen comillas pero si el programador ya las había puesto se interpretaba la sentencia como acabada y no la ejecutaba correctamente. El programador puede poner comillas simples, dobles o no poner nada en sus restricciones.

Para prevenir el problema se ha creado una función recursiva con una expresión regular que elimina las comillas simple y dobles del principio y final de la cadena dejándola lista para añadir las comillas por defecto. Dado que no se sabe cuando empezará el mensaje ni si será con comillas o sin sólo se eliminan recursivamente las comillas del tipo por el que empiecen.

EJ:

""""dani"""" → dani

""dani"" → 'dani'

""dani"" → "dani"

6.15.12 Buscar dinámico

No siempre nos interesa hacer consultas que nos devuelvan todos los campos, para solucionar esto se han utilizado métodos mágicos de php.

Se puede buscar cualquier campo de la Base de Datos simplemente poniendo "buscar" seguido del campo con la primera letra en mayúsculas o con un "_" seguido del nombre del campo, cualquier combinación de los casos anteriores es válida.

La función tiene como argumentos la restricción y el nombre de la tabla.

EJ: tabla Usuario, campos nombre, apellido e id

buscarNombreApellidoId("id=1", "Usuario");

buscar_Nombre_apellido_id("id=1", "Usuario");

buscar_nombreApellido_id("id=1", "Usuario");

En caso que simplemente se ponga "buscar" sin indicar ningún campo se interpreta como que se quiere el listado de todos los campos.

Para lograr esto se ha utilizado el método mágico __call, dicho método es llamado cuando se llama a una función que no está definida.

En dicho método se mira que el nombre de la función sea "buscar" o "find" y se aplican expresiones regulares para separar, en caso que sea necesario, los campos.

Después de tener todos los campos separados se hacen una serie de comprobaciones de seguridad y se genera la sentencia SQL con los datos necesarios.

6.15.13 Existencia de campos por los que buscar, separadores y contenido correcto

En el apartado anterior (8.1.12 *Buscar Dinámico*) hemos hablado de una serie de restricciones de seguridad.

En principio dado que la utilización de dicha función sólo será por parte de un usuario programador entendido en el tema se confiaba en el correcto paso de parámetros y elección de los campos correctos.

Después de utilizar estas funciones se da uno cuenta que es necesario dotar de información sobre el estado de la consulta al programador puesto que un error de comillas o de campos podría no retornar el valor adecuado o simplemente no retornar nada.

Con el siguiente ejemplo se pretende explicar el funcionamiento de los controles.

Ejemplo:

```
buscar_nombre('!=Dani');
```

Para empezar mediante una expresión regular se separan los campos de la restricción, como resultado obtenemos un array que contiene agrupaciones de 3 elementos, el primero el nombre del campo, el segundo la condición (=, !=, >, <, like, etc), y en tercer lugar el valor a comparar.

```
Campo1 = "!="
```

```
Campo2 = "Dani"
```

```
Campo3 = ""
```

Acto seguido se comprueba si el primer campo que debería ser el nombre del campo de la base de datos es un string o por lo contrario un símbolo de condición (=, !=, >, <, like, etc).

En caso que sea un signo de condición se comprueba que el número de parámetros de la función buscar sea igual a los de la condición.

La función buscar tiene un parámetro que es el nombre y la condición también tiene un parámetro ya que sólo hay un separador de condición.

Si los parámetros coinciden se sabe que la restricción es por el nombre del campo que se busca.

A la condición le falta el nombre del campo pero como coinciden el número de parámetros se sabe que ese campo es nombre, es decir

```
Campo1 = "nombre"
```

```
Campo2 = "!="
```

```
Campo3 = "Dani"
```

Después de esta comprobación podemos asegurar que si no había el nombre del campo ahora si lo hay.

Por lo tanto aseguramos bloques de 3 elementos para cada condición (nombre campo, separador condición, valor de la comparación).

Ahora hacemos saltos de 3 en 3, el siguiente campo a estos saltos tiene que ser un “and” o “or” para concatenar las restricciones, en caso que no lo sean se notifica el error. Además al hacer los saltos de 3 en 3 se aplica la función de comprobación de las comillas explicada en los apartados anteriores (8.1.11 *Comprobar comillas*).

Se ejecute correctamente o falle en alguno de los pasos anteriores se informará al programador mediante un archivo de log.

6.15.14 Ejecución de código no deseado

Toda aplicación informática debe ser robusta y comprobar que todos los campos necesarios están rellenos y sin caracteres que pudieran alterar el correcto funcionamiento de la aplicación.

En un primer momento se controlaron todos los formularios, evitando campos necesarios vacíos y la inyección de SQL.

Después de estos controles se probó poner en los campos que permiten escribir fragmentos de código. La inyección SQL estaba controlada pero no la ejecución de código malicioso ni de ataques XSS.

Para solucionar esta inseguridad se aplicaron dos funciones que se encarga de limpiar los campos.

La primera “basicClean” se encarga de errores de codificación, de eliminar los elementos “on (clic, submit, etc)”, también de borrar código javascript, visual basic script, namespace, expresiones anidadas a css y palabras tabú de html (applet, meta, frame, iframe, etc).

La segunda “removeXSS” se encarga de codificar caracteres que no son imprimibles, caracteres en blanco, palabras reservadas como javascript, <?, <%, todos los eventos de javascript etc. Coge todas estas palabras reservada y le añade un tag en medio para partir su ejecución.

Además de estas dos funciones se aplica también la función “trim” para eliminar espacios innecesarios.

El resultado de todos estos controles se para por la función “strip_tags” para eliminar todas las etiquetas html y por último se le aplica la función “htmlspecialchars” para convertir caracteres especiales como &, ‘, “, a su codificación en html.

Ejemplo:

Tabla 1.1: Codificación

Carácter	Codificación
'	'
&	&
"	"
<	<

6.15.15 Clase Modelo

La idea de la clase modelo es que sea una clase estática, que no se necesite crear un objeto para utilizarla.

Las funciones declaradas en la clase modelo son accedida con el operador "::" de esta manera haciendo `Modelo::nombreFuncion();` podemos llamar a la función que deseemos sin necesidad de haber creado ni instanciado ningún objeto de ella.

El problema surge porque la clase modelo utiliza métodos mágicos, en concreto utiliza el método "`__call`" que en este caso nos sirve para que en la función buscar se puedan declarar los campos que se quieren obtener.

Los métodos mágicos en la actualidad no pueden ser estáticos. A día de hoy la versión oficial de php es la 5.2.10, se está trabajando en la versión 5.3.0. Una de las mejoras de la versión 5.3.0 es que da soporte para poder llamar a los métodos mágicos de manera estática. Actualmente se está probando la versión 5.3.0 RC4, pese a que ya está disponible para su descarga, aún está en fase de pruebas. Se están corrigiendo bugs y aplicando parte de las funcionalidades que caracterizará la versión 5.3. Dado que no es una versión aún estable ningún servidor comercial la tiene instalada ni da la posibilidad de probarla.

Ahora en el servicio Web para compartir plazas de aparcamiento se utiliza una mezcla. Todas las funciones se utilizan de manera estática excepto la función buscar que se utiliza creando el objeto Modelo.

Se han mezclado los dos métodos para que una vez esté disponible la versión 5.3 de php sea más fácil de cambiar estos métodos mágicos.

El acceso a la Base de Datos siempre se realiza desde el controlador de cada sección, dichos controladores heredan de la clase controlador.

En el constructor de la clase controlador está declarada la inicialización del objeto Modelo.

El día que se cambie todo el Modelo a estático simplemente se tendrá que eliminar el objeto modelo del constructor del controlador y cambiar en todos los controladores las instrucciones "`$this->m->`" por "`Modelo::`".

También se tendrá que eliminar el objeto modelo del “FrontControler” que no hereda del controlador.

Con estos simples cambios quedará toda la clase modelo de forma estática y no se tendrá que instanciar nunca más.

Capítulo 7

Migración

7.1 Migración de la Base de Datos

En la Base de Datos está toda información referente a los usuarios y las plazas de aparcamiento.

Dado que las tablas que existen en la actualidad son muy grandes resulta muy costoso descargarlas o tratarlas.

El volumen de información es tan grande que se necesitarían horas para descargarse, las consultas tendrían que ser parciales e ir anidando los resultados y otras cuantas horas más para volver a subir las tablas actualizadas.

La empresa que proporciona el servidor no permite acceso de root así que tampoco se pueden gestionar directamente por consola sobre el servidor, la gestión sobre la interfaz Web es muy limitada y lenta lo cual tampoco resulta viable.

El servicio va en el mismo servidor que está actualmente pero en un dominio diferente. Se ha pedido a la empresa encargada de la gestión del servidor que realice una copia de las tablas necesarias y las copie en el nuevo dominio.

La características de las tablas serán tratadas desde la interfaz gráfica de mysql cambiando sólo las propiedades necesarias y añadiendo los nuevos campos.

Los campos que no puedan estar a vacío son rellenados con NULL o 0 en caso que tenga que ser un número entero.

7.2 Migración de los ficheros de texto

En los ficheros de texto sólo hay dos tipos de información, los números identificadores de los clientes y la traducción de todos los textos.

En el caso de identificadores de clientes dado que hay menos de 10 se introducen los datos manualmente sobre la tabla Cobrandeds de la nueva Base de Datos.

En el caso de las traducciones se han revisado todos los textos y solamente se puede aprovechar una pequeña parte.

Las traducciones del sistema antiguo no abarcan todo el abanico de posibilidades del nuevo sistema, de tal manera se rescatara de manera puntual las traducciones que concuerden en ambos sistemas.

Dado que las traducciones no superan las 100 frases se seleccionaran las frases coincidentes de manera manual.

Capítulo 8

Conclusión

8.1 Conclusión

A continuación se exponen diferentes aspectos referentes al desarrollo del proyecto así como un listado de los objetivos cumplidos y una valoración personal.

8.1.1 Conclusión del proyecto

Debido a algunos problemas durante la programación se ha alargado la fase de desarrollo reduciendo tiempo para la realización de la parte final de la memoria.

Podemos decir que el proyecto se ha realizado satisfactoriamente, se han cumplido todos los objetivos y el cliente obtiene un producto novedoso y adecuado a las características solicitadas.

8.1.2 Objetivos cumplidos

La aplicación cumple todos los objetivos que se plantearon en el diseño, se han incluido mejoras como no sólo referenciar las plazas en el mapa sino que también se han enlazados los resultados para simplificar la visualización.

También se ha tenido en cuenta que el servicio continúe funcionando en caso de no tener javascript activado.

Se ha desarrollado también una librería para generar estadísticas utilizando la API de google chart, actualmente sólo se utiliza una de las funciones (gráficos circulares en 3D) pero está preparada para poder utilizar gráficos verticales, horizontales, de comparativa, de líneas, entre otros.

Aunque no se especificaba explícitamente se ha desarrollado un frameWork para facilitar la ampliación de la aplicación y para crear aplicaciones nuevas de manera más sencillas. La peculiaridad de este frameWork es que permite olvidarse de las sentencias SQL. Se han creados una serie de funciones como guardar, modificar, borrar y buscar que automáticamente generan las sentencias.

Este hecho combinado con el desarrollo de una librería específicamente desarrollada para interactuar con mysql dota a este frameWork de mucha simplicidad cuando se tiene que modificar. Puesto que hay sentencias que no son fáciles de predecir existe la posibilidad de crear un modelo específico que hereda todas las características anteriores y permite ampliar las funcionalidades.

Cabe indicar que las pruebas realizadas de la aplicación han sido superadas correctamente.

El cálculo del presupuesto no se ha visto superado al finalizar el proyecto.

8.1.3 Valoración subjetiva

Personalmente puedo confirmar que la aplicación está lista para el funcionamiento.

El desarrollo de la aplicación ha sido un reto personal.

Es la primera aplicación diseñada y programada íntegramente por mí para una empresa real.

El diseño y la documentación han sido campos nuevos para mí mientras que la programación en sí es algo que llevo desarrollando hace algunos años.

Creo que ha sido una experiencia positiva para valorar todos los aspectos de un proyecto.

El realizar el diseño y calcular los costes antes de empezar a desarrollar es algo que hasta ahora no se hacía y en este proyecto me he dado cuenta que es imprescindible.

Concluyo con una valoración positiva del trabajo realizado y de la metodología aplicada.

8.2 Líneas futuras

Dado que es un proyecto de desarrollo, desde que se definieron los requisitos hasta el día de hoy han ido ampliándose las necesidades para mantener la aplicación actualizada y al día.

Actualmente se han fijado nuevos requerimientos para ampliar la aplicación:

- Que el servicio sea compatible con teléfonos móviles.
- Que el idioma por defecto que salga al cliente sea dependiendo de la IP y no del navegador, actualmente hay usuarios con el navegador en inglés simplemente porque no saben cambiar el idioma.

Con el idioma por IP se evitaría este problema y se asignaría el idioma oficial para cada país o región del país.

- Incluir nuevas funcionalidades sobre el mapa, como por ejemplo al mostrar los resultados de la búsqueda añadir una opción en el globo de google maps dejando ver fotografías reales de la calle (StreetView)
- Crear un chat para que el hecho de vender, alquilar o intercambiar la plaza de aparcamiento sea más rápida y llevadera.
- Funcionalidades para poder rellenar los contratos de alquiler o intercambio de manera automática y poder descargarlos directamente para firmar. El simple hecho de rellenar estos contratos supone medidas adicionales de seguridad sobre los contratos en sí.

Existen también otras mejoras que aún están por estudiar y que podrían ser interesantes:

Bibliografía

PHP. Wikipedia.

Online: <http://es.wikipedia.org/wiki/.php> (2009, Febrero 15)

PHP. Wikipedia.

Online: http://es.wikipedia.org/wiki/Action_Mailer (2009, Febrero 15)

PHP. Wikipedia.

Online: http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado (2009, Febrero 15)

MySQL AB (2009). 10.10.1. Conjuntos de caracteres Unicode. Mysql.

Online: <http://dev.mysql.com/doc/refman/5.0/es/charset-unicode-sets.html> (2009, 17 febrero)

Mark Davis, Ken Whistler (28 marzo 2008). Unicode Collation Algorithm. Unicode.

Online: <http://www.unicode.org/reports/tr10/> (2009, 17 febrero)

Gilberto Stankiewicz (4 junio 2008). CHARACTER SET y COLLATION En MySQL. Stan.

Online: <http://www.stan.com.mx/topics/view/12> (2009, 17 febrero)

Luis Sacristán (16 junio 2006). Simular dominio en localhost con Apache. Sentidoweb.

Online: <http://sentidoweb.com/2006/06/16/simular-dominio-en-localhost-con-apache.php> (2009, Febrero 21)

William Wong Garay (27 marzo 2006). Manual básico de creación de Host virtuales en Apache. Maestrodelsweb.

Online: <http://www.maestrosdelweb.com/editorial/host/> (2009, Febrero 21)

The Apache Software Foundation (2009). VirtualHost Examples. Apache.

Online: <http://httpd.apache.org/docs/2.0/vhosts/examples.html> (2009, Febrero 21)

Omer Ganovic (20 mayo 2007). MVC con PHP. Phppizza.

Online: <http://www.phppizza.com/2007/05/20/mvc-con-php/es/> (2009, Febrero 22)

Bibliografía

Rasmus (27 febrero 2006). The no-framework PHP MVC framework. Lerdorf.

Online: <http://toys.lerdorf.com/archives/38-The-no-framework-PHP-MVC-framework.html> (2009, Febrero 22)

Alias "Cienfuegos" (23 diciembre 2007). MVC y PHP - Para programadores. Maringa.

Online: <http://www.taringa.net/posts/info/1003158/MVC-y-PHP---Para-programadores.html> (2009, Febrero 25)

Joaquin Gracia (27 mayo 2005). Diseño de Software Orientado a Objetos. Ingenierossoftware.

Online: <http://www.ingenierossoftware.com/analisisydiseño/patrones-diseno.php> (2009, Febrero 25)

Miguel-Angel Sicilia(25 noviembre 2008).Ejemplo de Ingeniería Inversa de Datos. Cnx.

Online: <http://cnx.org/content/m17436/latest/> (2009, Febrero 25)

J. Carlos Montalvo (Agosto 2008). Comenzando MVC con php. Calinsoft. .

Online: <http://www.calinsoft.com/2008/08/comenzando-mvc-con-php.html> (2009, Marzo 1)

Federico (19 septiembre 2007). Introducción a MVC con PHP, primera parte. Journmoly

Online: <http://www.journmoly.com.ar/introduccion-a-mvc-con-php-primera-parte/> (2009, Marzo 1)

Federico (19 septiembre 2007). Introducción a MVC con PHP, primera parte. Journmoly

Online: <http://www.journmoly.com.ar/introduccion-a-mvc-con-php-segunda-parte/> (2009, Marzo 1)

Factoría de Internet S.L (2003-2008). MVC en tres pasos, en PHP y sin objetos. Webtaller .

Online: <http://www.webtaller.com/construccion/lenguajes/php/lecciones/mvc-tres-pasos-php-sin-objetos.php> (2009, Marzo 1)

Bibliografía

Ing. Febe Ángel Ciudad Ricardo (Abril 2006). Utilización del Patrón Modelo – Vista – Controlador (MVC) en el diseño de software educativos - Monografias.com. Monografías .

Online: <http://www.monografias.com/trabajos43/patron-modelo-vista/patron-modelo-vista2.shtml> (2009, Marzo 11)

Washington Sosa (Septiembre 2007). Zend_Controller quick start (del manual de zend framework). evolucionphp. .

Online: <http://evolucionphp.blogspot.com/2007/09/zendcontroller-quick-start-del-manual.html> (2009, Marzo 12)

Christopher (Octubre 2007). FrontController. Web2development. .

Online: <http://web2development.blogspot.com/2007/10/frontcontroller.html> (2009, Marzo 12)

Enrique Place (Septiembre 2005). Patrón de Diseño "Singleton" .phpsenior. .

Online: <http://phpsenior.blogspot.com/2005/09/patrn-de-diseo-singleton.html> (2009, Abril 1)

Eloi de San Martin (5 Abril 2006). Sesiones PHP. Programacionweb. .

Online: <http://www.programacionweb.net/articulos/articulo/?num=377> (2009, Abril 12)

Quelli Coto (Enero 2008). Configurando apache con mod_rewrite. Noth. .

Online: http://www.noth.es/2006/06/21/configurando-apache-con-mod_rewrite/ (2009, Abril 15)

Diego Agulló (29 junio 2005). Reescribir direcciones con mod_rewrite de Apache. Aeoris. .

Online: http://www.aeoris.net/escritos/apache/reescribir-direcciones-con-mod_rewrite.html (2009, Abril 15)

Roberto Bárcenas (3 mazo 2004). Variables php. Desarrolloweb. .

Online: <http://www.desarrolloweb.com/articulos/1398.php> (2009, Abril 14)

Bibliografía

The PHP Group (19 junio 2009). get_class_vars. Php.

Online: <http://es.php.net/manual/es/function.get-class-vars.php> (2009, Abril 15)

The PHP Group (19 junio 2009). get_class. Php.

Online: <http://es.php.net/manual/es/function.get-class.php> (2009, Abril 15)

The PHP Group (19 junio 2009). call_user_func. Php.

Online: <http://es.php.net/manual/es/function.call-user-func.php> (2009, Abril 17)

Manuel Gutiérrez Heredia (9 diciembre 2007). Expresiones regulares en PHP. Tufuncion.

Online: http://www.tufuncion.com/expresiones_regulares_php (2009, Abril 19)

Blasten (24 septiembre 2003). Expresiones Regulares en PHP. Manualdephp.

Online: <http://www.manualdephp.com/codigos-php/expresiones-regulares-php.html> (2009, Abril 19)

Bartolomé Sintés Marco (31 marzo 2009). Expresiones regulares. PHP. Mclibre.

Online:

http://www.mclibre.org/consultar/php/lecciones/php_expresiones_regulares.html

(2009, Abril 19)

Irv (14 enero 2007). Expresiones Regulares. Iginside.

Online: <http://www.iginside.net/man/php/regex.php> (2009, Abril 20)

Irv (29 julio 2006). Una breve referencia de los meta caracteres y su significado en expresiones regulares en PHP. Desarrolloweb.

Online: <http://www.desarrolloweb.com/articulos/referencia-expresiones-regulares.html>

(2009, Abril 20)

Arias (2009). Manual de Expresiones Regulares :: Versión 1. php-hispano.

Online: <http://www.php-hispano.net/archivos/Manuales/212/1/Manual-de-Expresiones-Regulares1.html> (2009, Abril 20)

Pedro Hernández Arauzo (2009). Validación con expresiones regulares. Gollum.

Online: http://gollum.inforq.uniovi.es/aii/valida_regex.php (2009, Abril 25)

Bibliografía

Abdicar Communications, S.A. y AxiRED Internet Solutions Company, S.A (2000-2009). Describe la sintaxis de expresiones regulares PCRE en PHP, Manual Oficial de PHP. Hackingballz.

Online: <http://www.hackingballz.com/herramientas/manual-oficial-de-php/reference.pcre.pattern.syntax.html> (2009, Abril 25)

Miguel Angel Alvarez (22 febrero 2008). Patrones de expresiones regulares para htaccess. Desarrolloweb.

Online: <http://www.desarrolloweb.com/articulos/patrones-expresiones-regulares-htaccess.html> (2009, Abril 25)

Luis Sacristán (7 abril 2008). URL routing con PHP. sentidoweb.

Online: <http://sentidoweb.com/2008/04/07/url-routing-con-php.php> (2009, Abril 27)

The PHP Group (19 junio 2009). strtotime. Php.

Online: <http://es2.php.net/strtotime> (2009, Mayo 10)

Alias "oicram"(4 junio 2007). PHP General :: today date + 1 day. Interaktonline.

Online: <http://www.interaktonline.com/support/forums/php-general-forum/Details/137006/today+date+%2B+1+day..html?last=1> (2009, Mayo 10)

Alias "assgar"(18 enero 2008). increment date by week. Daniweb.

Online: <http://www.daniweb.com/forums/thread105292.html#> (2009, Mayo 10)

Google Maps API Samples (2009). Create a map from a published Google Spreadsheet. google.

Online: <http://gmaps-samples.googlecode.com/svn/trunk/spreadsheetsmapwizard/makecustommap.htm> (2009, Mayo 19)

Timothy Groves (2006-2009). Autosuggest / Autocomplete with Ajax. Brandspankingnew.

Online: http://www.brandspankingnew.net/specials/ajax_autosuggest/ajax_autosuggest_autocomplete.html (2009, Mayo 20)

Bibliografía

The PHP Group (19 junio 2009). SimpleXMLElement->xpath. Php.

Online: <http://es.php.net/manual/es/function.simplexml-element-xpath.php> (2009, Mayo 22)

Matej Lednar (7 mayo 2009).Node.firstChild. Mozilla.

Online: <https://developer.mozilla.org/En/DOM/Node.firstChild> (2009, Mayo 25)

HuttarI (1 agosto 2008).Traversing an HTML table with JavaScript and DOM Interfaces. Mozilla.

Online:

https://developer.mozilla.org/en/traversing_an_html_table_with_javascript_and_dom_interfaces (2009, Mayo 25)

The PHP Group (19 junio 2009). utf8_encode. Php.

Online: <http://es2.php.net/manual/es/function.utf8-encode.php> (2009, Mayo 25)

Pamela Fox (28 febrero 2008). Introducing the Maps API Demo Gallery. Googlemapsapi.

Online: <http://googlemapsapi.blogspot.com/2008/02/introducing-maps-api-demo-gallery.html> (2009, Mayo 28)

Google Maps API Examples and Test Pages. Mbnet.

Online: <http://koti.mbnet.fi/ojalesa/exam/index.html> (2009 Mayo 28)

Mike Williams (Agosto 2006).Mike's Little Web Page. Econym.

Online: <http://www.econym.demon.co.uk/> (2009, Mayo 28)

Mike Williams (Agosto 2006). Google Maps API Tutorial. Econym.

Online: <http://econym.org.uk/gmap/> (2009, Mayo 28)

Google (2009). Ejemplos del API de Google Maps. Google.

Online: <http://code.google.com/intl/es/apis/maps/documentation/examples/index.html> (2009, Mayo 28)

Bibliografía

Google (2009). Google Maps JavaScript API Example: LocalSearch Control. Google.
Online: <http://code.google.com/intl/es/apis/maps/documentation/examples/control-localsearch.html> (2009, Mayo 28)

Google (2009). Google Maps JavaScript API Example: Title Detector. Google.
Online: <http://code.google.com/intl/es/apis/maps/documentation/examples/tile-detector.html> (2009, Mayo 28)

Google (2009). Google Maps JavaScript API Example: Geocoding Simple. Google.
Online: <http://code.google.com/intl/es/apis/maps/documentation/examples/geocoding-simple.html> (2009, Mayo 28)

Google (2009). Google Maps JavaScript API Example: Geocoding cache. Google.
Online: <http://code.google.com/intl/es/apis/maps/documentation/examples/geocoding-cache.html> (2009, Mayo 28)

Google (2009). Google Maps API Example : GGeoXml RSS Overlay. Google.
Online: <http://code.google.com/intl/es/apis/maps/documentation/examples/geoxml-rss.html> (2009, Mayo 28)

Google (2009). Referencias para el API de Google Maps - API de Google Maps - Google Code. Google.
Online: <http://code.google.com/intl/es-ES/apis/maps/documentation/reference.html#GMarkerOptions> (2009, Mayo 28)

Google (2009). Servicios. Google.
Online: <http://code.google.com/intl/es/apis/maps/documentation/services.html> (2009, Mayo 28)

Google (2009). Google Maps API Example : Geocoding reverse. Google.
Online: <http://code.google.com/intl/es/apis/maps/documentation/examples/geocoding-reverse.html> (2009, Mayo 28)

Google (2009). Google Maps API Example - Google North America Offices. Google.
Online: http://gmaps-utility-library.googlecode.com/svn/trunk/markermanager/release/examples/google_northamerica_offices.html (2009, Mayo 28)

Bibliografía

Ayesha ali (16 mayo 2009). Google Maps API. Google.

Online: <http://groups.google.com/group/Google-Maps-API/msg/d8aaf8189e1f69b0?pli=1> (2009, Mayo 28)

Mike Williams (Agosto 2006). Google Maps API Tutorial. Econym.

Online: <http://econym.org.uk/gmap/gevent.htm> (2009, Mayo 30)

Mika Tuupola (Noviembre 2008). Introduction to marker clustering with google maps. Appelsiini.

Online: <http://www.appelsiini.net/2008/11/introduction-to-marker-clustering-with-google-maps> (2009, Mayo 30)

Artículos sobre la api de Google Maps. Mappas.

Online: <http://www.mappas.org/blog/> (2009, Mayo 30)

Alias "ikhuerta" (14 marzo 2009) Integrar Google Street View en nuestras webs. Ikhuerta.

Online: <http://blog.ikhuerta.com/integrar-google-street-view-en-nuestras-webs> (2009, Mayo 30)

Mike Williams (Agosto 2006). Modularized Overlays and Controls. Econym.

Online: <http://econym.org.uk/gmap/modularized.htm> (2009, Mayo 30)

Mike Williams (Agosto 2006). Sidebar. Econym.

Online: <http://econym.org.uk/gmap/basic2.htm> (2009, Mayo 30)

Mike Williams (Agosto 2006). XML. Econym.

Online: <http://econym.org.uk/gmap/basic3.htm> (2009, Mayo 30)

Mike Williams (Agosto 2006). Links and Images, etc. Econym.

Online: <http://econym.org.uk/gmap/basic6.htm> (2009, Mayo 30)

Mike Williams (Agosto 2006). Google Maps. Econym.

Online: http://econym.org.uk/gmap/example_map11a.htm (2009, Mayo 30)

Mike Williams (Agosto 2006). Google Maps MarkerManager. Econym.

Online: http://econym.org.uk/gmap/example_markermanager.htm (2009, Mayo 30)

Bibliografía

Mike Williams (Agosto 2006). Google Maps. Econym.

Online: http://econym.org.uk/gmap/example_maptips2.htm (2009, Mayo 30)

Mike Williams (Agosto 2006). Custom Tooltips - new method. Econym.

Online: <http://econym.org.uk/gmap/tooltips4.htm> (2009, Mayo 30)

Mike Williams (Agosto 2006). Google Maps Mesage. Econym.

Online: http://econym.org.uk/gmap/example_message.htm (2009, Mayo 30)

Mike Williams (Agosto 2006). Google Maps Ebubble. Econym.

Online: http://econym.org.uk/gmap/example_ebubble.htm (2009, Mayo 30)

Mike Williams (Agosto 2006). Google Maps Insert. Econym.

Online: http://econym.org.uk/gmap/example_einsert.htm (2009, Mayo 30)

Alias "Blade_Stom"(7 junio 2008). html entities vs strip_tags. Forosdelweb.

Online: http://www.forosdelweb.com/f18/html-entities-vs-strip_tags-594350/ (2009, Junio 2)

Gareth Watts (22 junio 2009). PHPXref 0.7: bbPress. Ftwr.

Online: <http://phpxref.ftwr.co.uk/bbpress/nav.html?bb-includes/functions.bb-formatting.php.source.html> (2009, Junio 2)

Christian Stocker(2001-2008). lx_externalinput_clean. Liip.

Online: <https://svn.liip.ch/repos/public/ext/externalinput/trunk/lx/externalinput/clean.php> (2009, Junio 2)

Alias "DomainManDave" (18 septiembre 2008). PHP Security Class. Namepros.

Online: <http://www.namepros.com/code/515465-php-security-class.html> (2009, Junio 2)

Meter (19 abril 2008). PHP sessions on shared hosting - Hack It!. Supernifty.

Online: <http://www.supernifty.com.au/blog/2008/04/19/php-sessions-on-shared-hosting-hack-3/> (2009, Junio 2)

Bibliografía

Seth Fogie (11 agosto 2006). XSS, Cookies, and Session ID Authentication – Three Ingredients for a Successful Hack. Informit.

Online: <http://www.informit.com/articles/article.aspx?p=603037&seqNum=1> (2009, Junio 2)

Daniel (10 octubre 2007). Ataques XSS: el peligro de hacer echo \$_GET['var']. Formatoweb.

Online: <http://www.formatoweb.com.ar/blog/2007/10/10/ataques-xss-el-peligro-de-hacer-echo-get-var/> (2009, Junio 2)

Eduardo García Alfaro(3 junio 2009).Uso de la clase PHPMailer. Programación.

Online: <http://www.programacion.com/php/articulo/phpmailer/> (2009, Junio 5)

Element index for package PHPMailer. Bblog.

Online: http://www.bblog.com/apidoc/elementindex_PHPMailer.html (2009, Junio 5)

Brent R. Matzelle(2001-2003) PHPMailer - PHP email class. Kodeers.

Online:

<http://www.koders.com/php/fid4F1BADA1A40AB0E33016206B2D5F24A03F0BDAC5.aspx?s=smt+server#L6> (2009, Junio 10)

Capítulo 8: Conclusión

- Posibilidad de utilizar el IDCat para poder registrarse y obtener los datos del contrato por este medio.
- Dado que se ha desarrollado un frameWork sería interesante adaptarlo de manera que fuera una API, ya sea privada para la empresa, de libre uso o de compra de licencias periódicas.